Expertise
and insight
for the future

Oanh Vo

Developing a System Measuring Temperature Dependence of Photodetectors

Metropolia University of Applied Sciences

Bachelor of Engineering

Degree Programme in Electronics

Bachelor's Thesis

07 May 2020

Author	Oanh Vo
Title	Developing a System Measuring Temperature Dependence of Photodetectors
Number of Pages	42 pages + 5 appendices
Date	07 May 2020
Degree	Bachelor of Engineering
Degree Programme	Electronics
Instructors	Markus Nenonen, Senior Engineer Matti Fischer, Principal Lecturer

This thesis work was commissioned by Emberion Oy. A measurement system for investigating the temperature dependence of Emberion photodetectors was needed. The goal was to develop a system consisting of instruments, measurement PCBs, and software, which enables measurement automation. Upon having a complete system, temperature measurements shall be conducted, followed by an analysis of measurement results.

A complete measurement system was successfully designed and implemented: temperature measurement daughterboard PCB was designed using PADS VX.2.4, data acquisitions and measurement automation were fully achieved by LabVIEW, and GNU Octave 5.2.0 was employed to perform data analysis. Emberion photodetectors were measured and the relationship between temperature and their performance was analysed. It was concluded that the generated photovoltage, hence the responsivity of a photodetector becomes better when its temperature decreases. Furthermore, an offset dark voltage within the photodetector, which is proportional to the temperature, was observed in the absence of light, and it has a direct effect on photosensitivity.

Understanding the temperature dependency of photodetectors provides insights for design considerations of high-performance photodetectors. Cooling was suggested as an efficient method to overcome the temperature limitation. It, however, exhibits other design challenges as it requires more costs, space, and power.

Keywords	Photodetectors, photodiodes, temperature dependence

Contents

List of Abbreviations

ADC	Analog-to-Digital Conversion
CAM	Computer Aided Manufacturing
CMOS	Complementary Metal Oxide Semiconductor
CSV	Comma-Separated Values
D	Specific Detectivity
DAC	Digital-to-Analog Conversion
DBL	Double-precision, Floating-point numeric value
DC	Direct Current
DLL	Dynamic Link Library
DMM	Digital Multi-Meter
DUT	Device Under Test
EMC	Electromagnetic Compatibility
FIFO	First-in First-out
FPGA	Field-Programmable Gate Array
GPIO	General Purpose Input/Output
G-R	Generation-Recombination
I/O	Input/Output
IR	Infrared
ND	Neutral Density (filter)
NEI	Noise Equivalent Irradiance
NEP	Noise Equivalent Power

NTC	Negative Temperature Coefficient (thermistor)
PC	Photoconductive
PCB	Printed Circuit Board
PID	Proportional, Integral, Derivative (controller)
PV	Photovoltaic
Q.E	Quantum Efficiency
R	Responsivity
RMS	Root Mean Square
ROIC	CMOS Read-out Integrated Circuits
SEQ	Sequence
SMB	Sub-Miniature version B (connector)
SMD	Surface-Mounted Device
SNR	Signal-to-Noise Ratio
SP3T	Single Pole Three Throw (switch)
TEC	Thermoelectric Cooling
VI	Virtual Instrument

1 Introduction

Examination of photodetector behaviours under different temperature conditions is crucial for the development of high-quality photodetectors at Emberion Oy. To achieve this, a temperature measurement system was required to be developed, which is the main goal of this thesis project. The development of this measurement system includes the design of a measurement daughterboard as well as software design for measurement automation and data acquisition. Additionally, conducting temperature measurements and performing results analysis were other targets of this project, as they are essentially important for investigating temperature dependency of photodetectors.

Emberion Oy is a spin-out company from NOKIA which was founded in 2016, Emberion differentiates itself by the unique value proposition of offering high-performance, wideband photodetectors with highly competitive cost. Cutting-edge imaging and sensing products of Emberion are developed thanks to strong nanomaterials processing expertise and its custom-designed CMOS ROICs (CMOS Read-out Integrated Circuits), together with hardware and software integration know-how. Emberion's products have a wide range of applications. They are typically used, for example, in industrial processes, quality inspections, material sorting sectors, security, surveillance, defence, rescue, automotive, medical applications, etc. [1.]

2 Theoretical Background

2.1 Photodetector Fundamentals

2.1.1 Photodetectors: an Overview

Optical detectors are devices which detect optical radiation and convert it into some electrical signals such as current, or voltage. They have two subcategories: *(1) thermal detectors* which sense, then convert radiative heat energy into an electrical signal, and *(2) photodetectors* which produce an electrical signal from detected light energy. [2,214.]

Only the most fundamental concepts of photodetectors are emphasized in this thesis, due to the very extensive, broad nature of optical detection field.

Photodetectors can be found everywhere – in commercial, residential, entertainment, research, biomedicine, industrial, military, and astronomy applications. They can be utilized in the ubiquitous TV remote controls, cameras, automatic door opening sensors, or high-speed IR fibreoptic communication systems. In safety, surveillance, security and defence fields, they can be sensitive IR motion sensors, more advanced collision detectors used in autonomous vehicles, or electrical fault, smoke, gas sensors. In material processing industry, different types of material are diagnosed and sorted thanks to sophisticated photodetectors of higher performance, accuracy and wider spectral response range. State-of-the-art photodetectors also play an important role in environmental sensing, medical imaging, thermography, biomedical analysis, spectroscopy, etc. [3.]

2.1.2 Photodetector Classification & Operation

Based on the nature of electric field development, photodetectors can be further subcategorized into two main groups: photoconductive (PC) and photovoltaic (PV) detectors.

Photoconductive detectors have a simple structure comprising of a slab of semiconductor such as silicon, and two ohmic contacts at two ends, as shown in figure 1a.

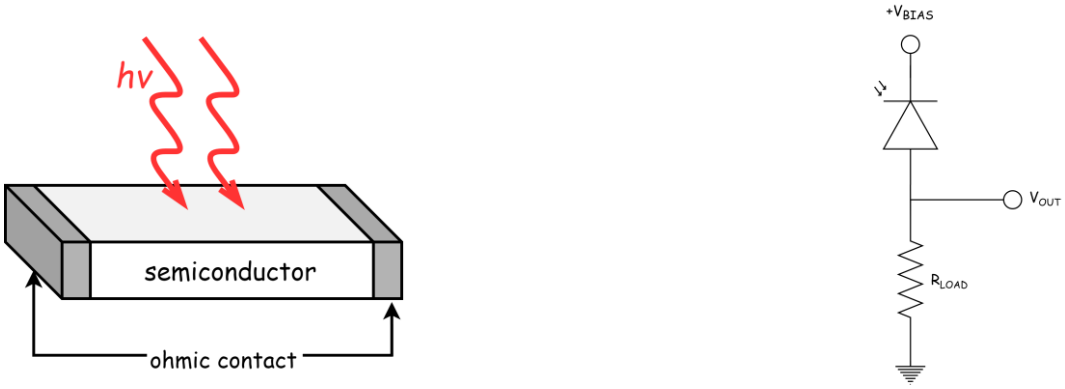


Figure 1. Left to right: 1a) Geometry of a PC detector. 1b) An example of a basic application circuit of PC detectors.

Arriving photons whose energy equal or greater than the bandgap energy of the semiconductor material (which is generated by supplying a bias voltage across the two electrodes) are absorbed, and they create excited electrons freely travelling in the photoconductor. The change in the number of electron-hole pairs in the PC device upon illumination gives the change in conductivity: the greater the incoming light intensity at a specific wavelength, the higher the generated current (or the corresponding voltage drop in the circuit configuration using a load resistor as shown in figure 1b). [2,224-225; 4,3-4.]

Photovoltaic detectors differ from PC detectors in the way the electric field separating electron-hole pairs is developed; meaning the electric field is internally induced by a p-n junction in PV detectors rather than being generated externally by a bias voltage in the case of PC detectors. This p-n junction is the interface between two semiconductor layers stacked together: one is p-type doped and the other one is n-type doped, as depicted in figure 2a.

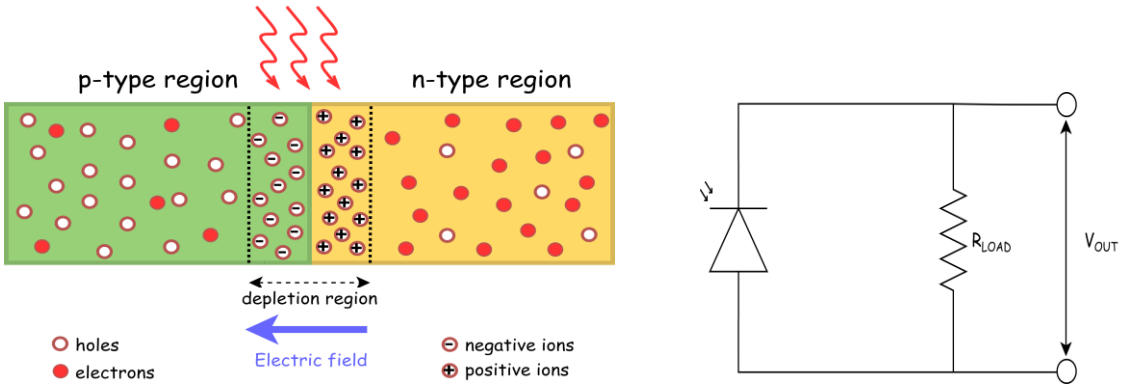


Figure 2. Left to right: 2a) A p-n junction. 2b) Basic circuit measuring PV detector output.

Upon illumination, arriving photons are absorbed in the depletion region of the p-n junction, which energy-boost the electrons to be able to escape from the valence band to the conduction band, thus producing electron-hole pairs. These electron-hole pairs are separated due to the built-in electric field created at the p-n junction in such a way that electrons and holes are drifted to the n-region and p-region respectively. This charge separation creates a photocurrent constituting a change in the voltage difference across the p-n junction (or generated photovoltage) which can be measured using external circuitry for example as in figure 2b. [2,222; 4,3-4.]

PV detectors are usually of more interest than PC detectors at the same material choice thanks to their better signal-to-noise ratio performance (up to $\sqrt{2}$ times lower noise level), more reliable predicted responsivity, and unbiased operation which gives better performance stability. On the other hand, PC detectors are often utilized in applications requiring higher light intensity and faster response times such as optical communication or signal processing. [5,157; 6.]

A popular variant of PV detectors is PIN photodiodes, which have enhanced frequency response and are the main photodetector type of interest in this thesis project, will be discussed in more details in section 2.2.

2.1.3 Performance Characteristics

The performance of photodetectors can be evaluated based on some key figures of merit, presented in the following paragraphs.

Responsivity implies detection sensitivity, i.e., how effectively a photodetector can convert light power received into electrical signal. Understanding the responsivity of a detector is beneficial, as the radiant flux level presented at a detector can be conveniently derived, as well as additional signal amplification, measuring, and processing circuitries can be more accurately planned to develop an optimal detecting system. The most fundamental concept of responsivity $\Re$ of detectors is demonstrated in equation (1) as the ratio of the output signal generated and the light signal input. Even though common unit for responsivity is [V/W], it can also be expressed in other unit terms such as [A/W], since output can be either photo current or photo voltage while input may represent radiant flux, optical power density, or optical total power. [5,41-43; 7,176.]

$$\Re = \frac{signal\ output}{incident\ ligh\ power} = \frac{V_{ph}\ (or\ I_{ph})}{P_{opt}} \quad (1)$$

One complicated, undesirable element that normally disturbs the performance of optical detectors is noise, because the signal to be detected is usually obscured, or masked by these random, unpredictable noise signals, which complicates the process of detecting, understanding, and processing signal output. In photonics applications where signals to

be detected are usually of small magnitude, attention to noise effects is critical. Noise may generate internally or externally. External noise sources, which are avoidable if the detection system is carefully planned, are for example, induced hum from the mains electricity, electric motor interference, static discharge or vibrations from outer environment, fluctuations in temperature, and so on. Noise coming from within the detection system itself, and is often inevitable, is internal noise, and is commonly categorized into three sources: Johnson noise, shot noise, and 1/f noise. [5,43; 8,117.]

Johnson noise (or *thermal noise,* or *white noise*), appearing at higher frequencies, is caused by the random, constant vibrations of electrons within a resistive device as a result of above absolute zero temperature (0 K). These constant movements create collisions not only between the charge carriers themselves, but also between the charge carriers and the atoms of the conducting material, producing minuscule currents which at a short time interval, are summed up to be Johnson noise current. The RMS noise voltage drop across the load resistor V_{jn} is usually expressed as

$$V_{jn} = \sqrt{4kTR\Delta f}\ , \tag{2}$$

or alternatively in RMS current form:

$$I_{jn} = \sqrt{\frac{4kT\Delta f}{R}}, \tag{3}$$

where k = 1.38 x 10-23 [J/K] is the Boltzmann constant, T [K] is the absolute temperature, Δf (or sometimes B) [Hz] is the bandwidth of the noise measurement, and R [Ω] is the load resistance. [2,221; 9,39-40.] From the straightforward relationship shown in equation (2), it is suggested by Ready [2,221] that lowering operating temperature by applying cooler to the detection system, reducing load resistor value and amplification bandwidth are necessary to reduce Johnson noise.

Shot noise, in contrast to thermal noise, is temperature and frequency independent (at low-intermediate frequencies) and only present when there is an external power source promoting the flow of charge carriers. Shot noise is described as “a consequence of the discrete nature of electric charge” [10,181], which implies the phenomenon of electrons

flowing to the anode at an inconsistent, random arrival rate, causing tiny fluctuations presented in DC current which constitute shot noise. Shot noise in semiconductors is referred as G-R (generation-recombination) noise as a result of the discrete rate of generating and recombining charge carriers. [2,221; 8,118; 10,181-182.] In order to reduce shot noise, Ready [2,221] advised lowering the magnitude of DC current as well as amplification bandwidth. The relationship of shot noise, current, and bandwidth is expressed in the equation calculating shot noise current I_{sn}

$$I_{sn} = \sqrt{2qI_D\Delta f}\ , \tag{4}$$

where I_D [A] is the DC current, q = 1.602 x 10-19 [C] is the electric charge, and Δf [Hz] is the noise measurement bandwidth [8,118; 9,41-42; 10,181-182].

1/f noise (*excess noise, flicker noise,* or *pink noise*) is named for its feature of having noise spectral density inversely proportional to the modulation frequency. In other words, at higher frequencies of 10^2 to 10^6 Hz, 1/f noise power can be neglected [11,205] and its interferences to electronic devices mainly stay at lower frequencies of few hundred Hertz [2,221]. The relationship of this noise and the modulation frequency is generally understood as $1/f^{\alpha}$ with α close to unity [8,118].

Apart from three main sources of internal noise coming from within the detection system discussed above, *photon noise* is a vital external noise factor which directly affects the detectivity parameter of the photodetector, and it is worth introducing. Naturally photons do not arrive at the detector at a constant rate due to the “discrete nature of the radiation field” [12,35], and it is further explained by Rogalski [12,35] that photons arriving from the target and those coming from the background constitute photon noise. The fluctuations in the arrival rate of photons from the source is inevitable while the background radiation noise can be managed for example, by lowering background temperature or by reducing the field of view [2,221-222].

Having briefly discussed some fundamental noise sources limiting the performance parameters of photodetectors, it can be concluded that it is crucial to maintain the noise level of one detection system at low at possible, beside having a high gain output signal,

as this helps detecting signals of low strength level be possible [8,666]. The next parameters are utilized to evaluate the noise performance of detectors.

Signal-to-Noise Ratio (SNR) is one parameter that characterizes the optical measurement quality, and is commonly presented as a ratio of detected optical power P_{signal} to noise power P_{noise}:

$$SNR = \frac{P_{signal}}{P_{noise}}, \tag{5}$$

or in an alternative form which describes the relationship between responsivity $\Re$, incident optical power P_{opt} , and noise current I_n [7,177]:

$$SNR = \frac{\Re\, P_{opt}}{I_n}, \tag{6}$$

where noise current I_n is the sum of individual noise components present in the detection system (I_{jn}, I_{sn}, etc.), as previously discussed [8,667].

It is emphasized by Vincent et al. [5,46] that even though the detector itself is not characterized by the SNR, the reliability of the optical measurement data is indeed reflected by the SNR, in other words, the higher the SNR, the stronger the data fidelity.

Noise Equivalent Power (NEP) which is associated with SNR, is another noise performance characteristic of a detector. NEP is defined as the required minimum incident optical power falling on a detector in order to yield SNR of one, or in other words, NEP is the optical power generating an output power (could be in form of current or voltage) of the same magnitude of the noise power. [7,177; 12,34.] The equation for calculating NEP is

$$NEP = \frac{I_n\ (or\ V_n)}{\Re}. \tag{7}$$

From the direct relationship between NEP, responsivity and noise power presented in equation (7), it can be concluded that a detection system performs better when NEP gets smaller.

Specific detectivity is another figure of merit presenting the sensitivity performance of a photodetector, or how well it can detect a signal. In its simplest form, detectivity is given as the reciprocal of the NEP value (D = 1/NEP). However, since using NEP value to compare the performance of different devices is only limited to those of the same photosensitive area, not to detectors of different geometries, a normalized detectivity or specific detectivity D* was introduced by Robert Clark Jones and is widely used. It is expressed as

$$D^* = \frac{\sqrt{A\,\Delta f}}{NEP} = \frac{\sqrt{A\,\Delta f}\,\Re}{I_n}, \tag{8}$$

where A represents the photosensitive area, Δf (or B) is the bandwidth, NEP is the noise equivalent power, $\Re$ is the responsivity of the device, and I_n is the noise current. The unit of D* is [cm $Hz^{1/2}$ W^{-1}] or [Jones]. Correspondingly, D* is also understood as the SNR of the detector of 1 cm^2 photosensitive surface detecting a light signal of 1W power in a bandwidth of 1Hz. This also indicates that the higher the D*, the better the detectivity performance of the detector. [7,177; 8,667; 12,34.]

Quantum efficiency, relating to responsivity, suggests how effectively a photodetector can convert photons radiating to the detector to photoelectrons, i.e., excited electrons generated from incident light. It is usually defined as a fraction of number of electrons produced divided by the number of incident photons:

$$Q.E = \eta = \frac{number\ of\ electrons\ produced}{number\ of\ incident\ photons}, \tag{9}$$

or formally in the relationship with responsivity:

$$Q.E = \eta = 100\ \Re\ \frac{1.2395}{\lambda}, \tag{10}$$

where $\Re$ is the responsivity and λ is the wavelength. [2,218.] It is obvious that a good detector is one with high quantum efficiency.

2.2 PIN Photodiodes

Recalling the operating principle of PV photodetectors: when light strikes an active region, photons are first absorbed in the depletion region, then electron-hole pairs are separated by the effect of built-in electric field, which results in the generated photocurrent or photovoltage. In order to make the electric field stronger thus improving the speed of photodetectors, a reverse bias voltage can be additionally supplied, meaning the positive terminal of the supply voltage is connected to the n-type semiconductor layer (cathode) while the negative supply terminal is applied to the p-type material (anode), as presented in figure 3. Furthermore, figure 3 also shows an important improvement made to the p-n junction architecture by introducing a wider layer of lightly doped, nearly intrinsic semiconductor material between a relatively thinner, heavily doped p-type layer and a n-type layer, hence the name ***PIN photodiode***. [2,227-228; 13,367-368.]

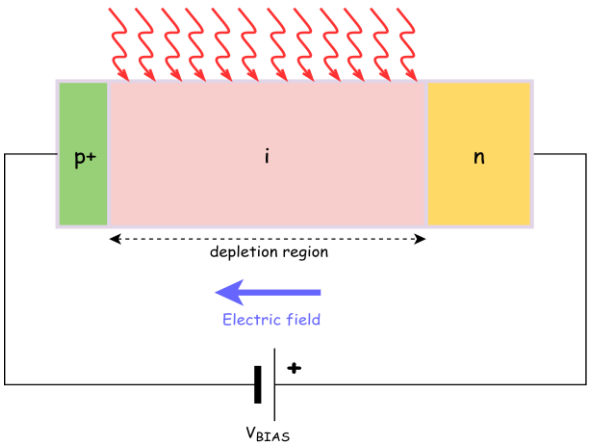


Figure 3. Reverse bias PIN photodiode.

PIN photodetectors have a similar light detecting mechanism as in p-n junction photosensors. However, in photo-detecting applications, PIN photodetectors are more of interest as they are much more effective than p-n junction photodiodes for many good reasons. First, the expansion in the depletion region width constitutes a larger light capturing area, thus maximizing the absorbance of incident photons, which leads to a *higher quantum efficiency η*. In normal p-n junction photodetectors, the doping concentration on both p and n side strongly affects the width of the depletion region. The higher the doping level, the narrower the depletion width, the less photons will be absorbed. On the

other hand, if the p and n regions are lightly doped, the conductance of the contact regions will be lowered. Orton [13,367-368] pointed out that a silicon PIN photodiode can satisfy both requirements that cannot be simultaneously met in p-n junction devices: a wide depletion layer (i-layer of width 30 – 50 µm could ideally give a Q.E yield close to unity) while maintaining a high dopant concentration in contact layers (at the order of 10^{25} m^{-3}). Secondly, thanks to the larger intrinsic region width, the junction capacitance C_J observed in PIN photodiodes is much more reduced, thus giving a *higher response speed* of up to 10^{10} Hz. [2,227-228; 13,367-368.]

An ***equivalent circuit*** representing a photodiode is depicted in figure 4.

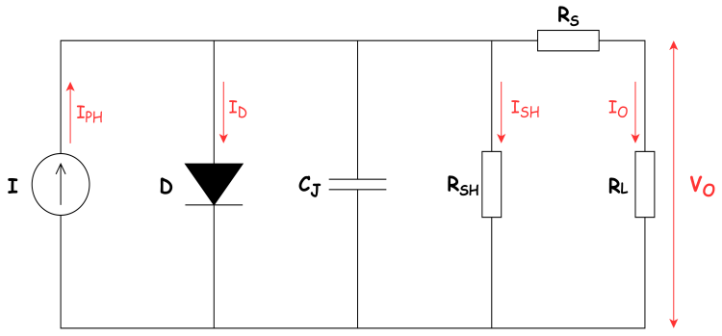


Figure 4. Equivalent model for a photodiode. I_{PH} is the current generated by the incident light power, D represents the p-n junction (or PIN junction), junction capacitance C_J and contact shunt resistance R_{SH} are in parallel while the series resistance R_S is in series. Load resistance is denoted as R_L. Figure adapted from [14,3].

Based on the equivalent circuit model in figure 4, using Kirchhoff's current law, the output current I_O upon illumination can be derived as:

$$I_{PH} - I_D - I_{SH} - I_O = 0 \tag{11a}$$

$$I_O = I_{PH} - I_D - I_{SH} \, . \tag{11b}$$

substituting the known Shockley diode equation $I_D = I_S \, (e^{\frac{qV_D}{kT}} - 1)$,

$$I_O = I_{PH} - I_S \, (e^{\frac{qV_D}{kT}} - 1) - I_{SH} \, , \tag{12}$$

where I_S is the reverse saturation current of the junction due to thermally generated carriers in the absence of light, V_D is the voltage across the junction, $q \approx 1.6 \times 10^{-19}$ [C] is the charge of electron, $k = 1.38 \times 10^{-23}$ [J/K] is the Boltzmann constant, T [K] is the absolute temperature.

A typical current vs. voltage, or ***I/V characteristics*** curve is usually found in photodiode datasheets, and an example is presented in figure 5. Applied voltage is on the horizontal axis while the current is on the vertical axis.

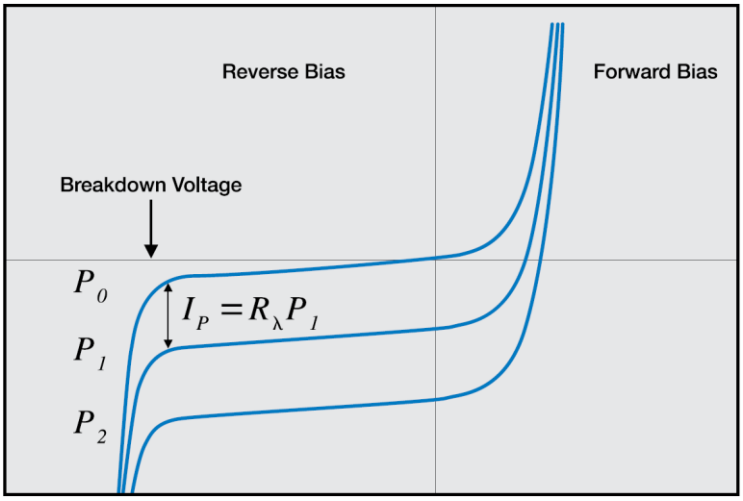


Figure 5. I/V characteristic of a photodiode. Reprinted from OSI Optoelectronics [14,5].

Under dark conditions (light level P_0), a photodiode behaves just as a p-n semiconductor diode, whose I/V relationship is shown in curve P_0 in figure 5. It has a Shockley diode current, as previously discussed $I_D = I_S\,(e^{\frac{qV_D}{kT}} - 1)$. When the diode is operating in forward bias mode, meaning when V = +V, the current is exponentially proportional to the applied voltage: $I_D = I_S\, e^{\frac{qV_D}{kT}}$. When a reverse bias voltage (V = -V) is applied, the exponential term becomes negligible, resulting $I_D = -I_S$. Upon illumination with light level P_1, in addition to the photodiode dark current, a reverse photocurrent is generated, and the I/V curve is shifted downwards, as shown in curve P_1. The shifted amount is the photocurrent I_{PH} (or I_P), which is the intersection of curve P_1 with the vertical axis. This relationship can be described as

$$I_O = I_S\,(e^{\frac{qV_D}{kT}} - 1) - I_{PH}\,. \tag{13}$$

Similarly, when a light of power $P_2 > P_1$ coming at a photodiode, the curve will be further downwards shifted, which is shown in curve P_2. When the reverse bias voltage gets high enough, it will eventually enter the breakdown region, where I_D will dramatically increase. This point is called breakdown voltage, and photodiodes should be operating below this limit. Photodiodes used for light detecting applications are operating in the third quadrant of the I/V curve (meaning the reverse bias region) while the four quadrant is meant for solar cells or energy harvesting applications. [14,5; 15.]

2.3 Temperature Effects on Photodiodes

In an ideal world, when there is no light coming to photodiodes, there should be no current. However, in real life, there is always a small amount of current flowing through the junction due to thermally generated carriers, which is known as **dark current**. It has been shown that this photodiode dark current is strongly dependent on the temperature. Grundmann [16, 416] pointed out that the magnitude of current in the junction (or photodiode dark current) increases as temperature rises, which is mainly due to "the temperature dependence of the saturation current". As previously shown in equation (13) in section 2.2, the output current I_O of a photodetector comprises of two components: (1) a non-zero dark current $I_S\ (e^{\frac{qV_D}{kT}} - 1)$ in the absence of light and (2) an added photocurrent I_{PH} upon illumination. In light detecting applications, this dark current is primarily the *reverse saturation current* I_S, because photodetectors operate on the third quadrant of the I/V curve, or in other words, they operate in reverse-biased mode, which makes the exponential term become negligible. The reverse saturation current of a p-n junction is defined as

$$I_S = A_j\, q\, {n_i}^2 \left(\frac{D_p}{L_p\, N_d} + \frac{D_n}{L_n\, N_a}\right), \tag{14}$$

where A_j is the cross-sectional area of the p-n junction, n_i is the intrinsic carrier concentration, D_p and D_n are the diffusion coefficients for holes and electrons respectively, L_p and L_n are diffusion lengths for holes and electrons respectively, N_d and N_a are the concentration of donor and acceptor correspondingly. It has been revealed that the effects

of temperature on reverse saturation current is strongly governed by parameter ${n_i}^2$, which is given as

$$ {n_i}^2 = \left(\sqrt{N_C\, N_V}\; e^{-\frac{E_g}{2k_BT}}\right)^2 $$

$$ = 4\left(\frac{2\pi k}{h^2}\right)^3 {m_e}^{3/2}\, {m_h}^{3/2}\, T^3 e^{-E_g/k_BT} \qquad (15) $$

with $N_C = 2\left(\frac{2\pi m_e kT}{h^2}\right)^{3/2}$ and $N_V = 2\left(\frac{2\pi m_h kT}{h^2}\right)^{3/2}$. N_C, N_V are effective density of states in the conduction and valence band, respectively; h is the Planck's constant, m_e and m_h are effective mass of electron and hole, and E_g is the bandgap energy. [10,43-45; 16,412-416; 17,5-7.]

From equations (14) and (15), the temperature dependence of the reverse saturation current can be estimated in the form

$$ I_S = C\, T^3 e^{-E_g/k_BT}, \qquad (16) $$

in which C is a constant including the terms which are either independent of temperature or very weakly temperature dependent which can be ignored. By differentiation, the change of saturation current for each temperature change is

$$ \frac{dI_S}{dT} = 3CT^2 e^{-\frac{E_g}{k_BT}} + \frac{CTE_g e^{-\frac{E_g}{k_BT}}}{k} $$

$$ = I_S\left(\frac{3}{T} + \frac{E_g}{k\,T^2}\right). \qquad (17) $$

Equation (17) can also be evaluated as

$$ \frac{d}{dT}\left[ln(I_S)\right] = \frac{1}{I_S} \times \frac{dI_S}{dT} = \frac{3}{T} + \frac{E_g}{k\,T^2} \qquad (18.1) $$

$$\Delta[ln(I_S)] = \Delta T \left(\frac{3}{T} + \frac{E_g}{k\,T^2}\right). \quad (18.2)$$

From this relationship, a general approximation rule has been used to evaluate the temperature dependency of saturation current, or in the case of photon detectors, dark current: for each 8 – 10K increase in the junction temperature, the reverse saturation current doubles. [17,5-7; 18, 64-65.]

As a result of the strong temperature dependence of dark current, breakdown voltage of photodiodes also decreases as temperature increases, meaning it becomes easier for the photodiode to enter its breakdown region. Finally, since the output of a photodetector is the sum of dark current and photo generated current, output of the photodetector is consequently dependent on the temperature, which means all the performance parameters of photodetectors would be greatly affected such as responsivity, Q.E, and noise parameters.

In photodetection technology, it is common to apply coolers to improve the performance of photodetectors. It has been believed that photosensors, especially those working in the longer wavelength region, generate more noise at room temperature due to the effects of thermally generated charge carriers. Cooling for photodetectors can be categorized into two mechanisms: (1) using liquid or solid cryogen packed in vacuum-insulated flasks called dewars to prevent cryogen from boiling away and (2) using refrigerators of types such as open or close-cycle refrigerators, radiation coolers, and thermoelectric coolers (TEC). Even though cooling is clearly a simple solution to the temperature dependence limitation of photodetectors, it still exhibits many impractical disadvantages when having coolers in a detection system. These are, for example, adding costs, weight and design complications to the detection system: the sensor should be carefully designed to be mounted to the cooler, the cooler should not stand in the way of incoming light to the detector, power consumption considerations, and so on. Ideally, a detector should be operating without the need of cooling, but for most applications, it is not yet the case. More efficient, economical coolers with better energy are also very beneficial for detection systems. [5,177; 19,3-10.]

3 Hardware Design

In order to measure the temperature dependence of one R&D batch of Emberion photodetectors, printed circuit boards (PCB) were naturally crucial beside all the measurement tools and instruments existing at Emberion's laboratory. This section mainly describes the designing process of the measurement daughterboard which was made specifically for measuring photodetectors in TO-8 cans. The measurement system can be introduced briefly in figure 6.

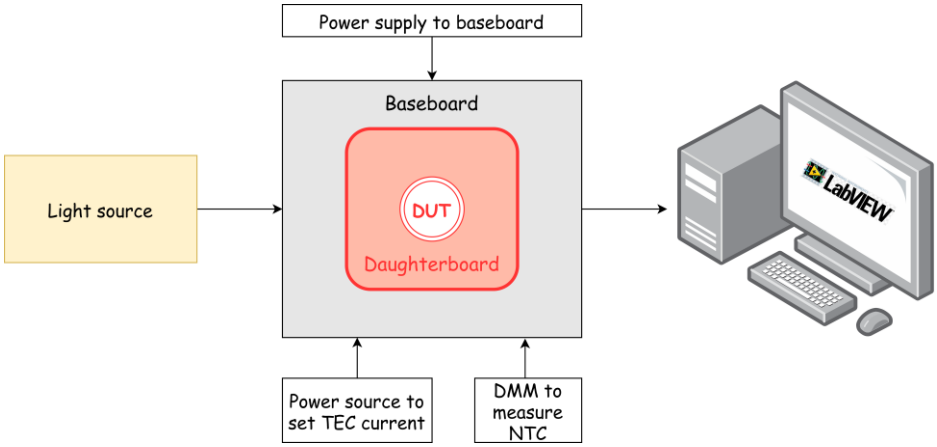


Figure 6. Measurement setup.

Figure 6 represents the measurement setup used for characterizing photodetectors. The device under test (DUT) was placed on the measurement daughterboard. This measurement daughterboard was then placed on Emberion baseboard via pogo connectors, thus supplies and signal lines were interconnected between baseboard and daughterboard. Baseboard is the core measurement board currently used at Emberion laboratory for characterizing different types of R&D photodetectors. For each specific generation of photodetectors, a specific daughterboard is designed accordingly. A light source system (including a lamp, harmonics filter, monochromator, neutral density filters, and an integrating sphere) was used to generate different light levels at different wavelengths. The DUT gave output signal which would be then acquired by LabVIEW, giving data saved in .CSV format for further analysis using MATLAB or Octave. Furthermore, a low noise power source was used to supply thermoelectric cooling (TEC) current in order to heat up or cool down the DUTs. In order to measure the current temperature in the device

using NTC thermistor, a digital multi meter (DMM) with the four-wire resistance measurement configuration was used.

3.1 Daughterboard Schematic

Starting the process of creating a daughterboard PCB, a logic schematic was designed for the daughterboard using ***PADS Designer VX.2.4***. The schematic for this baseboard consisted of two sheets. The first one represented the connections between baseboard and daughterboard, ID number of the daughterboard, and mechanical parts. These are illustrated in the figure 7.

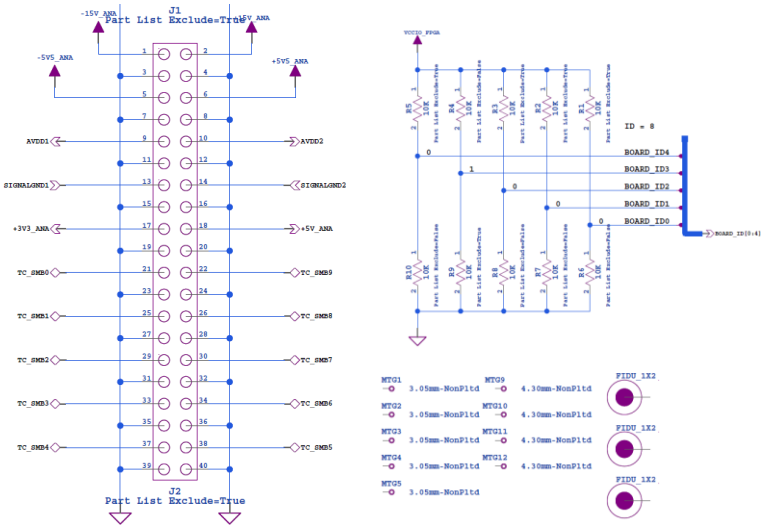


Figure 7. Some sections in the first sheet of the schematic

From figure 7, an example of an array of 40 pogo connections between daughterboard and baseboard is shown on the left. There were four of these arrays in the daughterboard. In the top right section, it shows the logic of how the ID of this specific daughterboard (which is 8 in decimal, or 1000 in binary) was generated. A network of pull-up and pull-down resistors was employed: pull-up resistors while excluding the placement of pull-down resistors gives a high logic, and vice versa. The lower right corner of figure 7 shows other important mechanical parts of the daughterboard, where mounting holes of different sizes were considered as it is crucial that the daughterboard must stay aligned with the baseboard.

The second sheet of the schematic contained the functional I/O connections between DUT photodetectors and measurement daughterboard. The DUTs were photodetectors containing four quadrants, each has 8x8 pixels (or totally 16x16 pixels) which were packaged in TO-8 cans, as depicted in figure 8.

Figure 8. Emberion R&D photodetectors packaged in TO-8 cans which have 12 I/O pins.

There are twelve I/O pins in these TO-8 can photodetectors, whose pin functions are presented in table 1. The twelve connections between the DUT and the daughterboard are further presented after that.

Table 1. I/O pin functions

TO-8 pins	Functions
1	V_{TG} – Top gate voltage
2	V_{OUT} – Output voltage of selected pixels
3	Row<n>
4	NTC+
5	NTC-
6	AVDD & AVDD_amp
7	SGND_ON
8	SignalGND
9	Pix<n>
10	TEC+
11	Chassis GND & AVSS & BG
12	TEC-

V_{TG} – top gate supply voltage: as indicated in figure 9, this pin was connected to a supply of 1.48 V with decoupling capacitors of values 100 µF, 10 µF, and 100 nF in parallel to reduce noise. The top gate voltage could be supplied either from VTG1 signal of the baseboard, or from SMB connector.

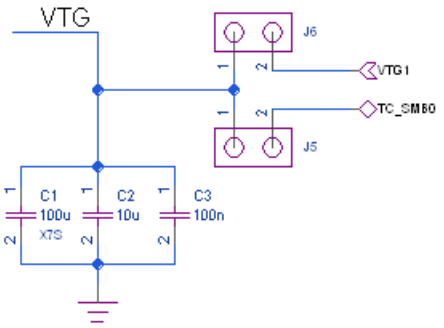


Figure 9. Top gate voltage circuitry

AVDD & AVDD_amp is a 2.2 V voltage supply pin to DUT, which was supplied either through baseboard or SMB connector, connecting with bypass capacitors 100 µF, 10 µF, and 100 nF in parallel. It has a similar circuitry as in figure 9.

SignalGND, which can be seen from figure 10, was a voltage signal supplied to DUTs either through baseboard or SMB connector with a voltage of 1.48 V. Multi bypass capacitors of values 100 µF, 10 µF, 100 nF, and 1 nF in parallel were applied, and a Zener diode MMBZ5226BLT1G was added to protect the DUT from oversupply.

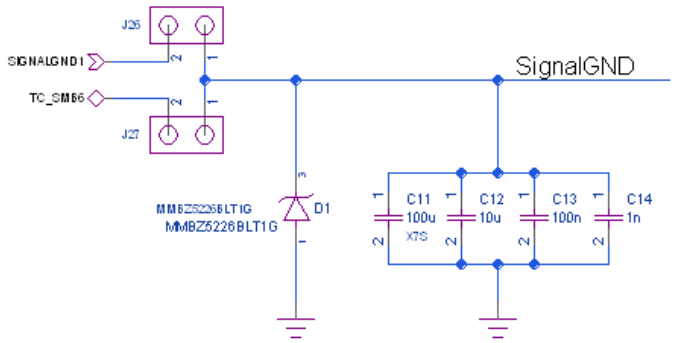


Figure 10. SignalGND circuitry

Row<n> and **Pix<n>** were digital signals used for pixel selection, whose circuitry is presented in figure 11. Because only four pixels were measured in this project, four combinations of high/low states of the two pins row<n> and pix<n> would enable the selection of those four interested pixels. Users could either set the row or pixel (or column) numbers manually or automatically from the FPGA code via GPIO2 and GPIO5 pins (more details about automatic row/column switching by FPGA presented in section 4.1). The selection was enabled by SP3T switches SS14MDP2: terminal one and three corresponded to high or low states set manually, and terminal two gave the digital signal set automatically from FPGA. Furthermore, voltage translators, or level shifters SN74LVC2T45 were used to shift the digital voltage level from FPGA (1.8 V) to the desired 2.2 V. Decoupling capacitors of 100 nF were used to filter supply noise and 10 kΩ pull-down resistors were used to maintain the default logic state to low, thus avoiding floating signals.

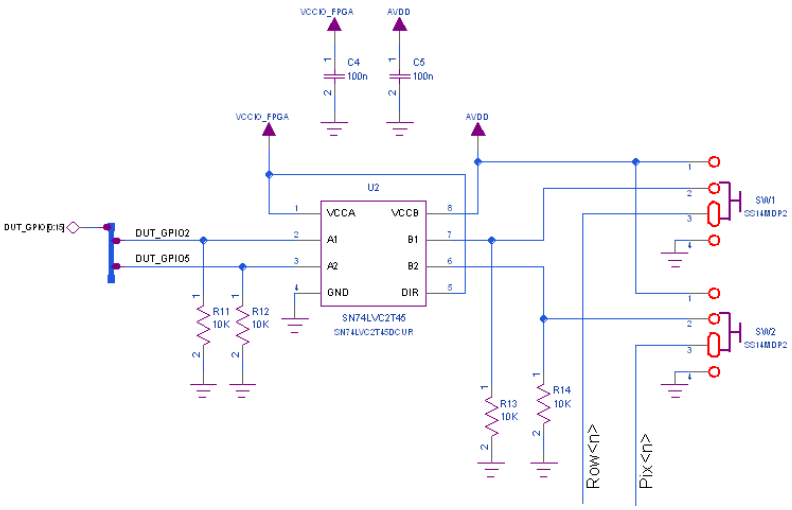


Figure 11. Row<n> and Pix<n> circuitry

SGND_ON signal is used for setting the pixel in reset or exposure state, and it could be set either manually of automatically by the FPGA. Voltage translators, decoupling capacitors and pull-down resistors were similarly used as in the case of row/pixel selection described above.

V_{OUT} – output voltage of selected pixels: its schematic is introduced in figure 12. Output voltage pin was amplified by a non-inverting operational amplifier LTC6240HV

configuration with the gain of 2. Supplies to op-amp were filtered by bypass capacitors of 100 nF.

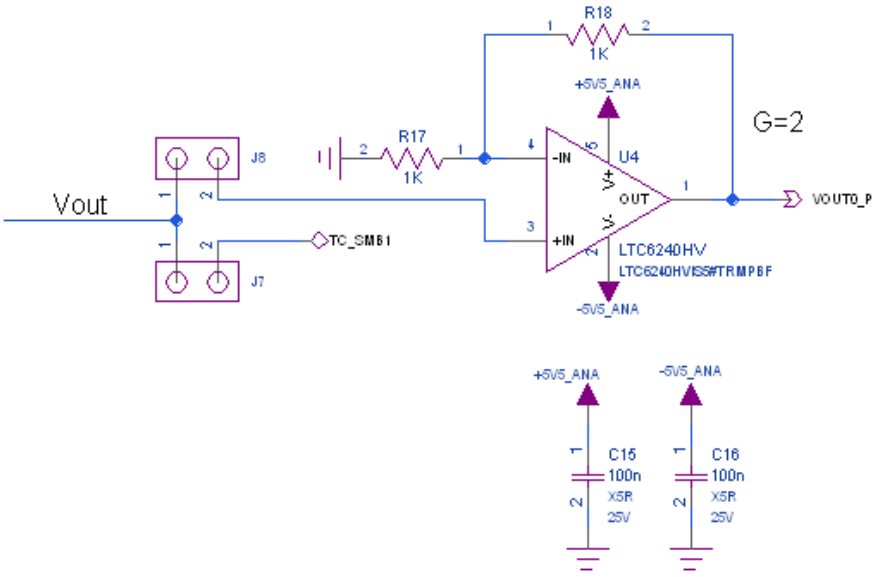


Figure 12. Amplified output voltage circuitry.

Chassis GND & AVSS & BG pin was grounded.

NTC- and **NTC+** (Negative Temperature Coefficient thermistor) pins were connected to a DMM using four-wire configuration as shown in figure 13. Four-wire configuration enables precise measurement of the thermistor resistance by eliminating cable resistances from the measurement. These two pins were connected to the DMM via two SMB cables (named NTC Force and NTC Sense), each SMB had a negative and a positive terminal, hence the notations NTC-_F, NTC+_F, NTC-_S, NTC+_S.

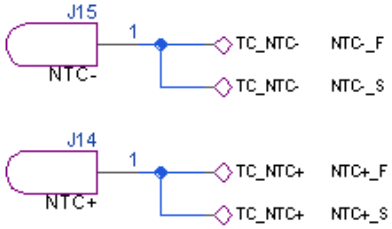


Figure 13. Four-wire measurement connection to measure NTC

TEC- and **TEC+** (Thermo Electric Cooling) pins as illustrated in figure 14, were connected to the corresponding negative and positive terminals of one SMB cable which would be then plugged into a low noise power supply source for setting the current needed to heat up or cool down the DUTs to a desired temperature.

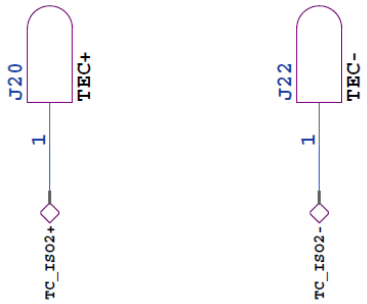


Figure 14. TEC connections

The components chosen in this daughterboard design were mostly SMDs. It is important to note that even though the schematic design was made in a way that there were two operating options for users, the desired way was automation using the baseboard, while adding a manual option was for debugging purposes only. A detailed schematic of this measurement daughterboard can be found in Appendix 1.

3.2 Daughterboard Layout

Upon having a complete, reviewed logic schematic for the daughterboard, a layout design was done using ***PADS Layout VX.2.4***. Logic schematic from PADS Designer was first forwarded to PADS Layout with relevant information of all the parts. Next, the components were tactfully arranged, partitioned, and placed, bearing in mind the coming routing requirements as well as user experience. In this project, many SMD filtering capacitors were used, thus it was important to place those bypass capacitors as close as possible to the power pins in order to reduce inductance and maximize decoupling performance. Furthermore, capacitors of smaller values were placed closer to the pins while the bigger capacitors could stay further as they were less critical. This useful and common PCB design tactic can be visually seen in figure 15.

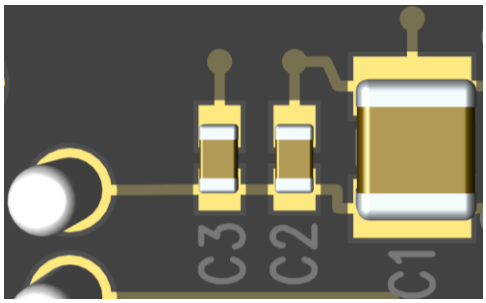


Figure 15. A small PCB section of the daughterboard in 3D view showing the bypass capacitors placement tactic applied in the daughterboard layout design.

Components placement in general were spanning out from the centre of the board, and certain distance was kept from mounting holes and keep-out zones (such as the areas reserved for hold down clamps) to prevent mechanical stress or component damage.

A recommended design rule was set and followed, which helped increasing the efficiency of layout design process while maintaining the crucial design requirements. These rules were for example, trace width, clearance between traces, vias, pads, drills, etc. There were four electrical layers used in this daughterboard PCB design: top and bottom layers for components placement and signals, inner layer two as a ground plane, and inner layer three as a mixed power plane. Routing of this daughterboard was done manually by starting with critical traces such as power and high-speed signals. Digital and analog traces were separated as much as possible while the reserved ground plane in inner layer two helped ensuring the shortest, lowest impedance current path to ground. Furthermore, traces were placed carefully to reduce trace length and sharp corners were avoided. Trace width of signals requiring significant current flow like TEC current traces were made to be wider to reduce the resistance and to ensure more current would be conducted, as shown in figure 16.

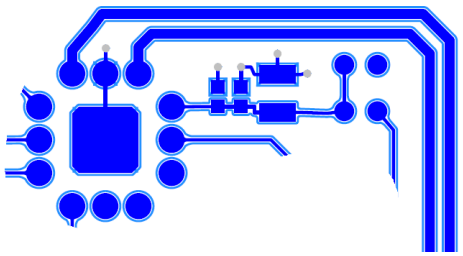


Figure 16. Trace width for TEC current supplies was made wider than other signal traces

Additionally, some power signals were made as power planes whenever possible within inner layer three, whose area were designed as large as possible, as displayed in figure 17. The pink area represents net VCCIO_FPGA (1.8 V digital), orange area is net AVDD (2.2 V digital), green and yellow regions show analog supplies of -5.5 V and +5.5 V respectively. The remaining area in blue was assigned as ground net.

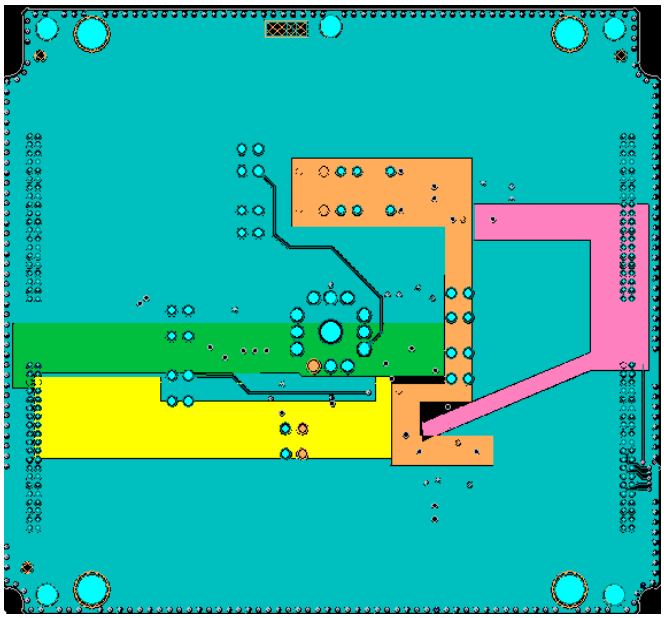


Figure 17. Inner layer three: mixed power planes

EMC matters were partly considered by assigning ground net to all the electrical layers, and by adding stitching vias around the daughterboard for better shielding as well as impedance and return loops reduction. The via stitching around the board can also be viewed from figure 17 around the perimeter of the daughterboard.

Finally, considering the user experience when using or debugging the daughterboard, it was necessary to add descriptive texts, labelling in the layer Silkscreen Top. 2D lines and Text functions from Drafting Toolbar were used to perform this task. Company logo, name and code for the daughterboard were also positioned mindfully. Mounting holes were placed accurately to ensure daughterboard would stay aligned with the baseboard after manufacturing. A complete view of the layout design can be visualized using PADS Layout 3D view, and it is presented in figure 18.

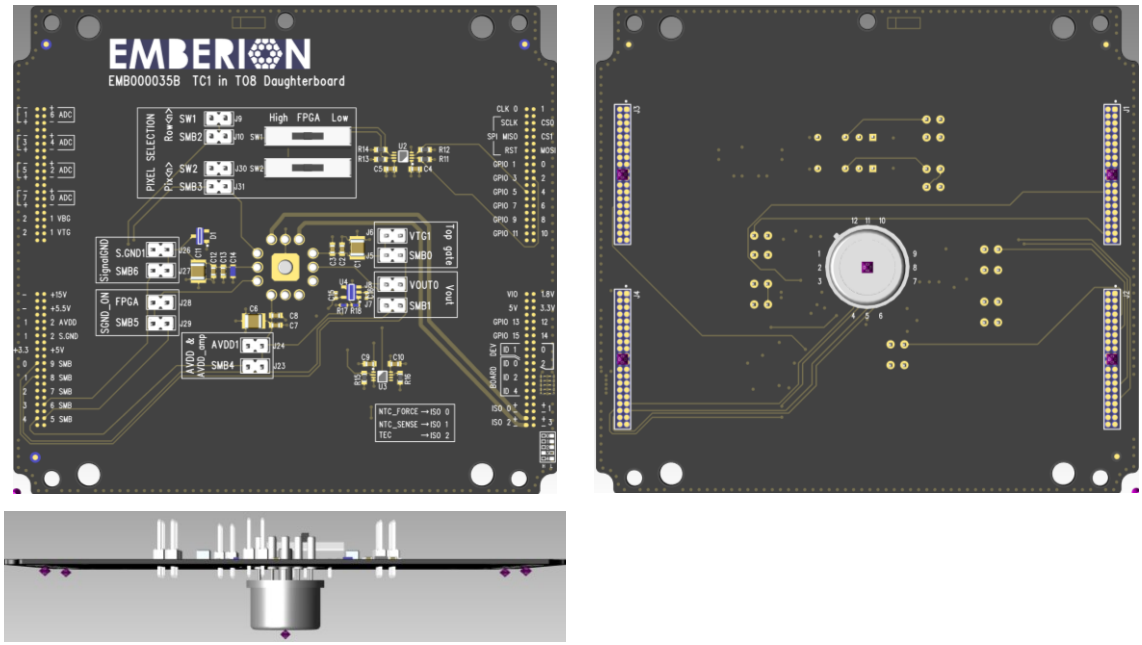


Figure 18. Daughterboard 3D view from left to right and from up to down: Top view, Bottom view, Side view

As depicted in figure 18, this measurement daughterboard had all the electronic components residing on the top side, while the bottom side was reserved for only placing the DUTs (in the middle) when conducting measurements. The dimensions of this daughterboard were width 9 cm x length 10 cm. Alternatively, the complete layout design of the daughterboard can also be viewed in its traditional graphical 2D presentation, which can be found in Appendix 2.

Once the layout design was ready and reviewed, the manufacturing Gerber outputs containing design information were created and delivered to PCB manufacturers. From PADS Layout, by using the CAM function, output documents can be previewed, defined and generated. The actual PCB is presented in figure 19.

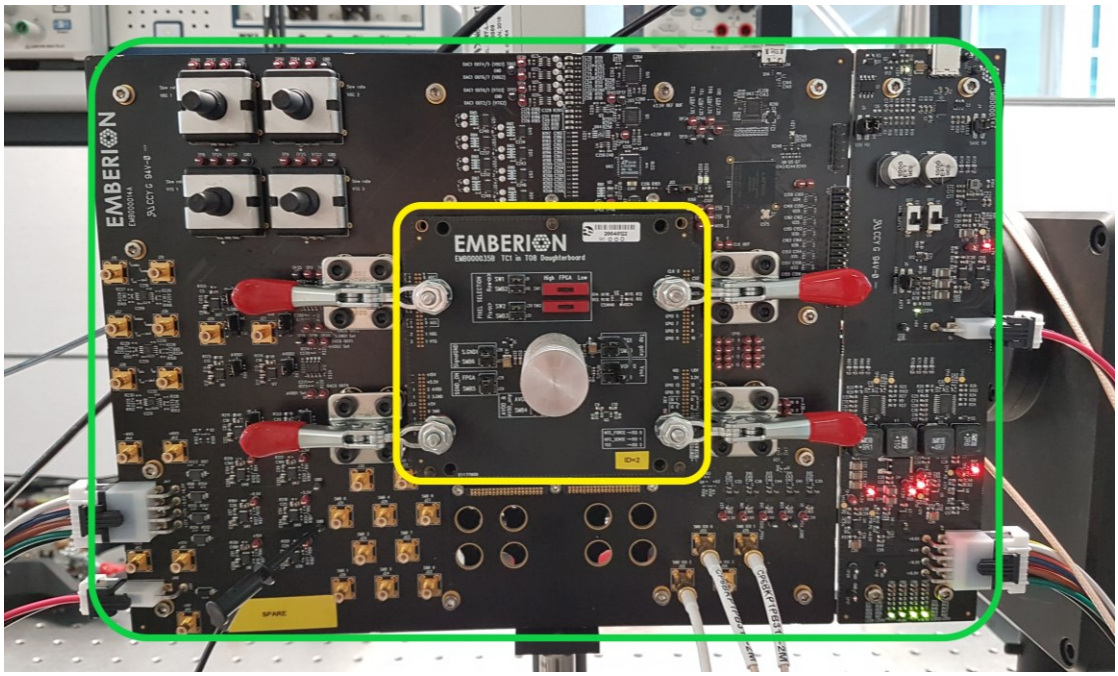


Figure 19. Baseboard and daughterboard in the measurement setup

As can be seen in the figure 19, the actual daughterboard PCB (marked yellow) stayed properly aligned with the baseboard (mark green). The pogo pin arrays were intact, and the four hold-down clamps gave a firm, strong lock between daughterboard and baseboard. In the middle of the daughterboard, a metallic heatsink can be seen and on the other side of the heatsink was the DUT.

4 Software Design

The current test frameworks established at Emberion laboratory are very extensive and they are the results of many years of development. For this reason, the scope of this thesis, specifically software design part, was initially defined that only a small fraction of temperature measurement functionalities for photodetectors in TO-8 package would be built based on existing frameworks. This includes (1) building a test sequence (.SEQ file) that enables measurement automation, (2) some small additions to the current LabVIEW test frameworks, and (3) a slight modification in the FPGA pixel sequencer to achieve the desired pixel switching. To illustrate the software elements contained in the current Emberion test framework and their relations, a block diagram is introduced in figure 20.

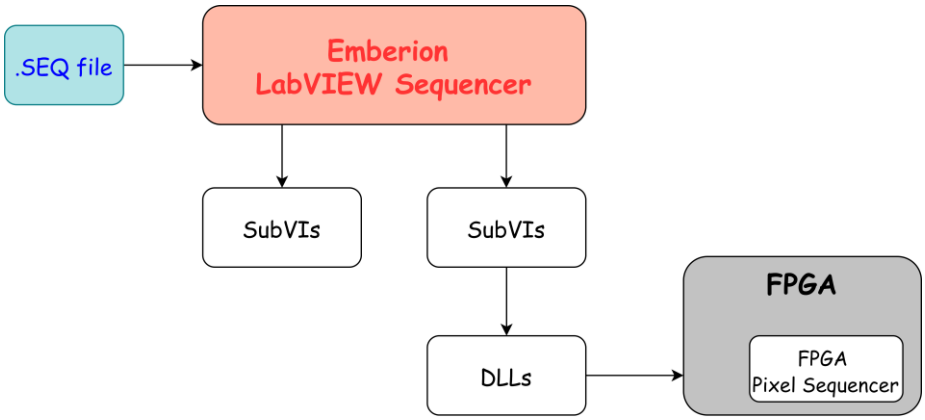


Figure 20. Block diagram of software elements

As can be seen in figure 20, the LabVIEW Emberion Sequencer consists of many subVIs, which are developed either by using LabVIEW built-in libraries or by assessing FPGA functions via Dynamic Link Libraries (DLLs). The development goal was that, by executing the .SEQ file commands from the Emberion Sequencer, users can call the desired functions (i.e., subVIs), as well as setting necessary parameter values of the FPGA pixel sequencer.

4.1 Test sequence

As can be seen from the block diagram of software elements presented in figure 20, a test sequence file (.SEQ file) plays a vital role in giving command to Emberion Sequencer program in order to enable measurement automation. This test sequence file consisted of four sub-sequences representing four different measurements: optical power sweep, wavelength sweep, reset & exposure time sweep, and temperature & power sweep. However, as the scope of this thesis was to study the temperature dependence of photodetectors, only the most relevant sub-sequence, which is ***temperature & power sweep*** and its corresponding ***dark signal measurement*** will be discussed.

The sequence started by prompting user for DUT information such as ID number, width, and length. Next, instruments were initialized. The hardware needed for this type of measurement was a Keysight B2962A power source to supply TEC current, a tunable light source system TLS260, a stepper motor controller ConexPP (for controlling different ND-filter positions which corresponded to different light intensity levels), a Keysight

34470A DMM to measure NTC, and a TEC PID controller. The sequence made to enable these tasks are illustrated in figure 21. The sequencer processed this .SEQ file line by line. In order to perform each functionality or to automatically drive each instrument, a corresponding VI was first addressed, after that, the specific desired function within that VI was called with the corresponding parameters and their values.

```
1  Ask User Info.vi        ask.string displayText(Enter Sample ID) dataName(SampleId)
2  Ask User Info.vi        ask.string displayText(Enter pixel width in um) dataName(width)
3  Ask User Info.vi        ask.string displayText(Enter pixel length in um) dataName(length)
4
5  //################### INITIALIZE HW ########################
6
7  //################### KEYSIGHT B2962A POWER SOURCE TO SET TEC CURRENT ############################
8  Keysight B2962A.vi      init    alias(tecSupply) address(USB0::0x0957::0x9018::MY52350827::INSTR)
9  Keysight B2962A.vi      configure.output    alias(tecSupply) enable(0) channel(channel1) complianceCurrent(0.3)
10 Keysight B2962A.vi      set.output.current  alias(tecSupply) channel(channel1) current(0)
11 Keysight B2962A.vi      configure.output    alias(tecSupply) enable(1) channel(channel1) complianceCurrent(0.3)
12
13 //################### ND-FILTER CONTROLLER #################
14 ConexPP.vi              init    com(com9)
15 ConexPP.vi              home.search
16 ConexPP.vi              move.absolute position(310)
17
18 //################### TUNABLE LIGHT SOURCE #################
19 TLS260.vi               init
20 TLS260.vi               set.lamp OnOff(On)
21 TLS260.vi               set.shutter openclose(Open)
22
23 //################# KEYSIGHT 34470A TO MEASURE NTC ###########################################################
24 Keysight 34470A.vi      init address(USB0::0x2A8D::0x0201::MY54700660::INSTR)
25 Keysight 34470A.vi      configure.measurement measurement(Resistance 4-wire)
26 Keysight 34470A.vi      configure.resistance integrationTime(0.02)
27
28 //################ TEC PID ###################################################################################
29 TEC PID Controller.vi   set.tec.limits maxCurrent(0.3) minCurrent(-0.3)
30 TEC PID Controller.vi   set.pid.gains gainP(0.1) gainI(0.03) gainD(0.1)
31 Timer.vi               wait time(2000) showUi(yes)
32 TEC PID Controller.vi   start.temperature.control targetT(0)
33 Timer.vi               wait time(5000) showUi(yes)
```

Figure 21. Test sequence: hardware initializations.

Next, the software designed to drive the baseboard and daughterboard was initialized, supply voltages were set from 12 DAC channels and reset/exposure time and ADC sampling time were defined. The command lines for this is represented in figure 22. The interested supplies in this project were 2.2 V AVDD (channels 0-1), 1.48 V SignalGND (channels 2-3), 1.48 V VTG (channels 4-7). A wait time of 2000 ms was added after these voltages were set. ADC sampling time was set to be 0.1 ms, reset time was 50 ms and exposure time was 100 ms. A delay of 5000 ms was also added.

```
36  8x8.vi          init
37  8x8.vi          set.analog.output channel(0) voltage(2.2)
38  8x8.vi          set.analog.output channel(1) voltage(2.2)
39  8x8.vi          set.analog.output channel(2) voltage(1.48)
40  8x8.vi          set.analog.output channel(3) voltage(1.48)
41  8x8.vi          set.analog.output channel(4) voltage(1.48)
42  8x8.vi          set.analog.output channel(5) voltage(1.48)
43  8x8.vi          set.analog.output channel(6) voltage(1.48)
44  8x8.vi          set.analog.output channel(7) voltage(1.48)
45  8x8.vi          set.analog.output channel(8) voltage(0)
46  8x8.vi          set.analog.output channel(9) voltage(0)
47  8x8.vi          set.analog.output channel(10) voltage(0)
48  8x8.vi          set.analog.output channel(11) voltage(0)
49  Timer.vi        wait time(2000) showUi(yes)
50
51  8x8.vi          set.pixel.sample.time sampleTime(0.1)
52  8x8.vi          set.reset.time resetTime(50)
53  8x8.vi          set.exposure.time exposureTime(100)
54  Timer.vi        wait time(5000) showUi(yes)
```

Figure 22. Test sequence: software initializations.

The sub-sequence ***temperature & power sweep*** started with the creation of a .CSV file for recording results data, and headers for these DUTs in TO-8 cans setup were added (which will be discussed in section 4.2). Next, the temperature of the DUT was up-swept from 30°C to 60°C, with step ΔT = 10°C; then it was down-swept from 50°C to 0°C with ΔT = -10°C; and finally up-swept again from 10°C to 30°C with ΔT = 10°C. In other words, the DUT was going through a temperature cycle: 30°C, 40°C, 50°C, 60°C, 50°C, 40°C, 30°C, 20°C, 10°C, 0°C, 10°C, 20°C, and 30°C.The target temperature was regulated using TEC and TEC PID controller. At each set temperature, the ***dark signal measurement*** of the DUT was first recorded at 500 times. After that, the DUT was measured under different optical power levels. This was achieved by moving the position of the ND-filter from 0 (brightest) to 310 (darkest), with the step of 10. The wavelength for this measurement was fixed at 1550 nm. At each light intensity level of a certain DUT temperature, result data was read 50 times and was saved into the result file. The full details of this temperature & power sweep sub-sequence can be found in Appendix 3.

4.2 LabVIEW Programming for Temperature Measurement Features

A comprehensive Emberion LabVIEW sequencer project for testing photodetectors had been built prior to the start of this thesis project. For this thesis project, making some additional features and small modifications were necessary in order to accommodate the changes in this specific generation of photodetectors.

First, functions (or subVIs) to define reset time and exposure time were added. The goal was to set those parameters from the .SEQ file, and these would define the corresponding values in FPGA pixel sequencer.

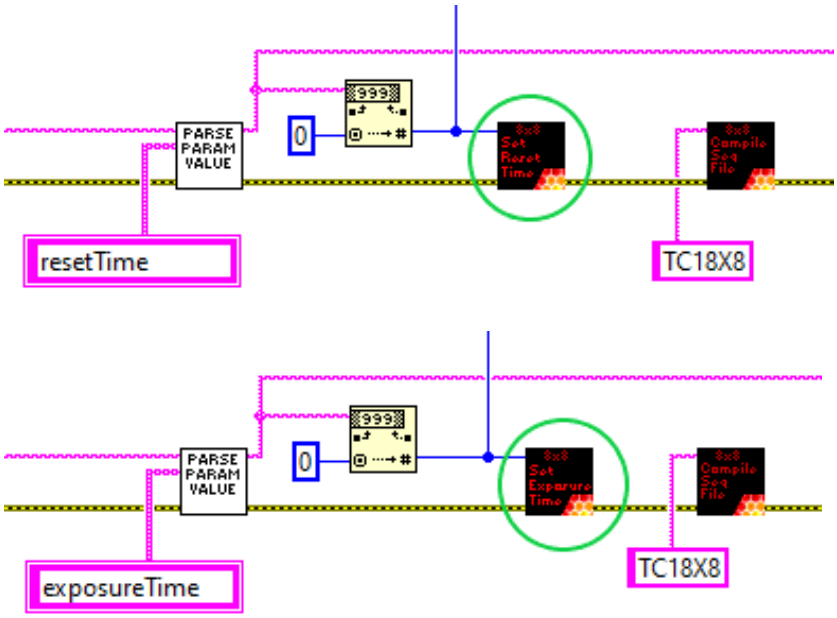


Figure 23. Data flow of set reset/exposure time functions

Figure 23 shows a small piece of code that describes the general data flow when `Set Reset Time.vi` and `Set Exposure Time.vi` (subVIs marked in green circle) were used within case structures in a VI of higher-level hierarchy. From the test sequence, users can define reset and exposure periods [ms]. These parameter values (`resetTime` and `exposureTime`) would be then parsed, extracted, and further converted to a U32 number. This number would be the input of the subVIs `Set Reset Time.vi` or `Set Exposure Time.vi`, whose code is presented in figure 24. Once the reset or exposure time was set, the FPGA pixel sequencer file would be compiled into a memory file for further executions.

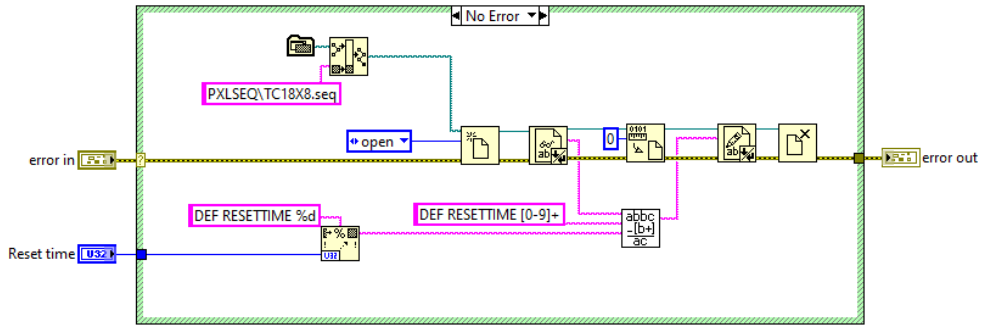


Figure 24. Block diagram of `Set Reset Time.vi`.

As shown in figure 24, the FPGA pixel sequencer file whose path was automatically defined, was first opened and read. As soon as a reset time was inputted, it was formatted into an output string containing a `RESETTIME` parameter with its defined value. This string would then replace the matching expression in the FPGA pixel sequencer, and this updated content would be written into the FPGA pixel sequencer file. At the end, the file was closed. Error handling was considered throughout every step of this VI: once an error happens, the sub-sequential function or subVI will not execute. The code designed was identical in the subVI `Set Exposure Time.vi`, only parameter `RESETTIME` would be then replaced by `EXPOSURETIME`.

Next, some modifications were made to properly read the measurement data of this batch of photodetectors whose only 4 out of 256 pixels would be read. Figure 25 displays a code segment whose functionality was data reading.

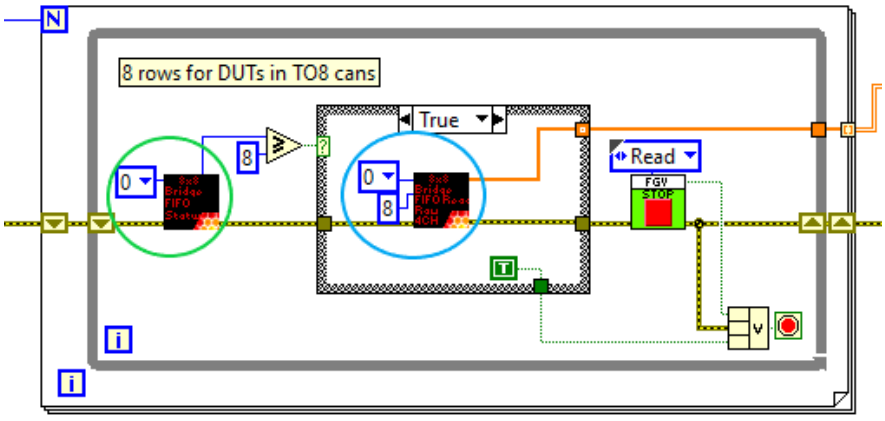


Figure 25. A section of LabVIEW block diagram for reading measurement data.

As shown in figure 25, the current status of the bridge FIFO was extracted by `Bridge FIFO Status.vi` (circled in green). FIFO channel 0 was set, and the extracted number of available rows in this FIFO channel was compared to 8 (i.e, the measurement results of four pixels during reset and exposure time). The result of this comparison decided the case selection of the case structure: data reading would happen when the FIFO had at least eight available rows. `Bridge FIFO Read Raw 4CH.vi` (circled in blue) was used to get the raw data from FIFO channel 0 with maximum 8 entries. Pointer to data buffer was used to read the contents within the FIFO buffer. The output data was of 1D DBL array type, and after N iterations of the outermost For Loop, this data became a 2D DBL array which would be further processed and saved in .CSV files.

Finally, before saving any batch data, headers for data columns in .CSV result files must be created. For DUTs in TO-8 cans, a specific set of headers were made, as shown in figure 26.

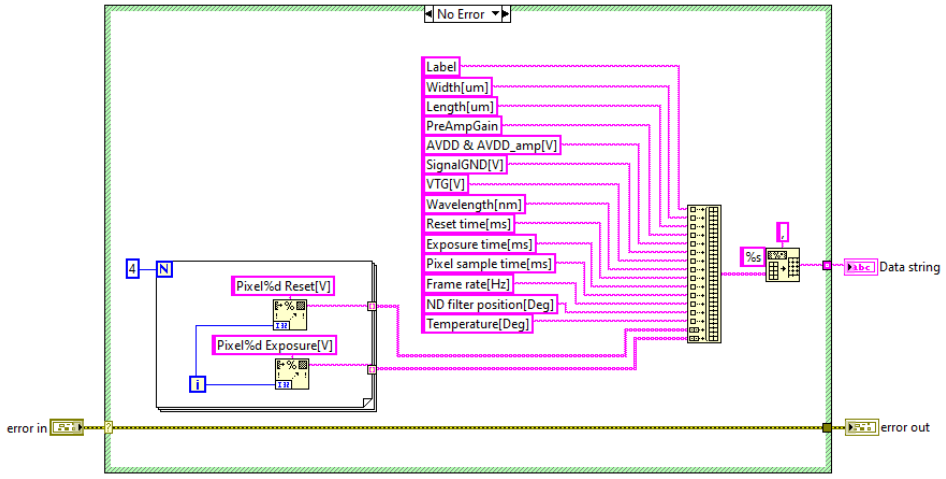
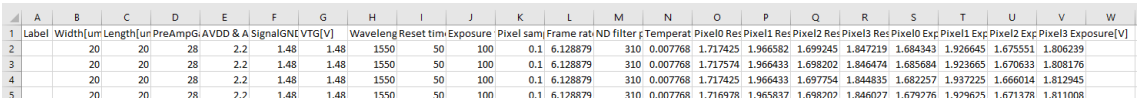


Figure 26. Headers created for DUTs in TO8 cans setup.

As presented in the LabVIEW block diagram in figure 26, an array of headers was built. The first 14 string-type headers i.e., Label, Width[um], Length[um], etc. were appended to make a 1D string array of 14 elements. Additionally, using a For Loop, headers of output voltages for four chosen pixels during reset and exposure were automatically created. After the execution of this For Loop, the output arrays would be further concatenated to build an end-result 1D array containing 22 string elements. This string array was finally converted to a .CSV spreadsheet of 22 columns.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
1	Label	Width[um	Length[un	PreAmpGa	AVDD & A	SignalGNI	VTG[V]	Waveleng	Reset tim	Exposure	Pixel sam	Frame rat	ND filter g	Temperat	Pixel0 Res	Pixel1 Res	Pixel2 Res	Pixel3 Res	Pixel0 Exp	Pixel1 Exp	Pixel2 Exp	Pixel3 Exposure[V]	
2		20	20	28	2.2	1.48	1.48	1550	50	100	0.1	6.128879	310	0.007768	1.717425	1.966582	1.699245	1.847219	1.684343	1.926645	1.675551	1.806239	
3		20	20	28	2.2	1.48	1.48	1550	50	100	0.1	6.128879	310	0.007768	1.717574	1.966433	1.698202	1.846474	1.685684	1.923665	1.670633	1.808176	
4		20	20	28	2.2	1.48	1.48	1550	50	100	0.1	6.128879	310	0.007768	1.717425	1.966433	1.697754	1.844835	1.682257	1.937225	1.666014	1.812945	
5		20	20	28	2.2	1.48	1.48	1550	50	100	0.1	6.128879	310	0.007768	1.716978	1.965837	1.698202	1.846027	1.679276	1.929625	1.671378	1.811008	

Figure 27. Column headers created in .CSV file

The data string would be later written in a result file, and the end result is depicted in figure 27. In the first row, from columns A to V, the headers were generated as a result of the LabVIEW code in figure 26.

4.3 FPGA Pixel Sequencer

As mentioned in section 3, the primary test approach was automation by having pixel switching and supply setting driven by FPGA built on the baseboard. An existing Emberion FPGA pixel sequencer program had been made for photodetector generation having 16x16 pixels in 4 quadrants (hence 8x8 pixels per quadrant). Pixel switching was done in such a way that the FPGA sequencer goes through every single pixel by looping through each row and column in each quadrant, samples the measured data, converts, then stores it in the memory. The row/column mapping for addressing different pixels in a quadrant is presented in table 2. Since only four pixels in one quadrant were to be measured in this project, it was necessary to modify the existing pixel sequencer which can perform switching of the four interesting pixels, which are marked **X** in Table 2.

Table 2. 8x8 pixels mapping in one quadrant

Row/ Column	0	1	2	3	4	5	6	7
0	X				X			
1								
2								
3								
4	X				X			
5								
6								
7								

Six digital lines were utilised to represent row and column number 0 – 7 in 3-bits binary form. The interested pixels were four combinations of rows **0**00, **1**00 and columns **0**00, **1**00. For this reason, it was enough to toggle only the MSBs (marked red) to address the desired pixels; this corresponds to digital lines 2 and 5. Moreover, a digital switch for reset/exposure of the sensor named SGND_ON was designed and digital line 6 was reserved for it.

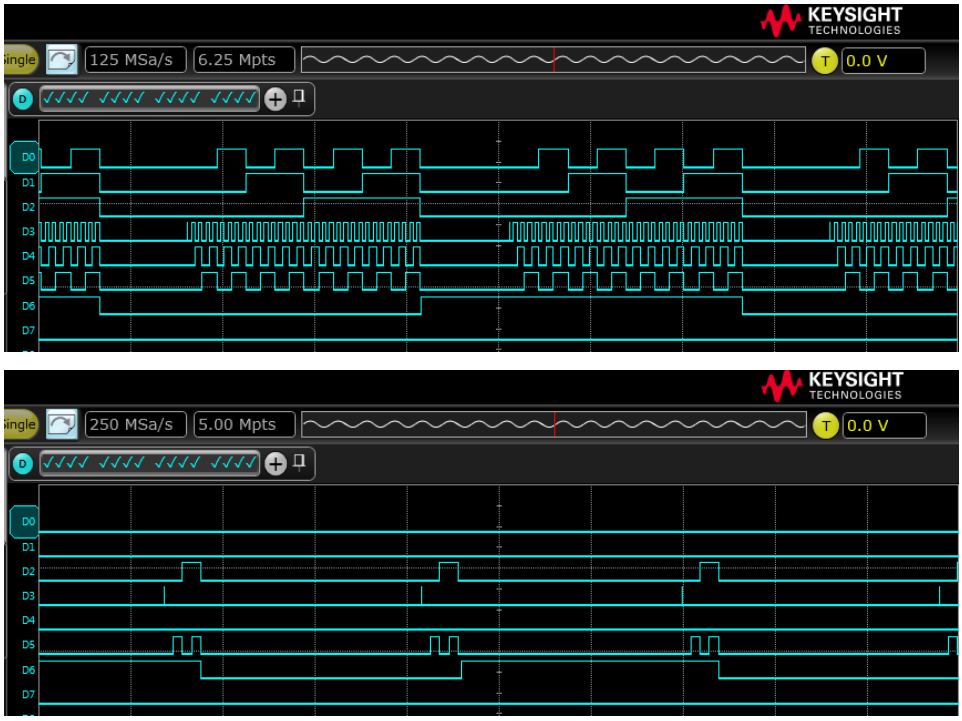


Figure 28. From up to down: 64 pixels switching, 4 pixels switching

Figure 28 further illustrates how the pixels switching was done by digitally measuring GPIOs using a Keysight oscilloscope. The upper picture shows all the six digital lines D0 – D5 were toggling for each reset (high) or exposure (low) period (D6), giving a combination of 64 different pixels accordingly. This was the result of the original, existing FPGA pixel sequencer. After the pixel sequencer was modified specifically for this project, the switching of four interesting pixels is reflected in the lower picture of figure 28. Only D2 and D5 were toggling during on/off time of D6, which corresponded to four chosen pixels.

5 Measurements & Results Analysis

An extensive Emberion-made LabVIEW sequencer had been used for automating the measurement process of photodetectors, meaning the test engineer only needed to place the DUTs in the daughterboard, and the rest of the measurement was driven automatically, following the .SEQ file. Figure 29 introduces the LabVIEW front panel of Emberion Sequencer. Users started the measurement by loading the measurement .SEQ file, start or stop the measurement by pressing start/stop sequence buttons. The sequencer executed the measurement .SEQ file line by line until the end of file or it will stop immediately once an error showed up. Information of the whole measurement system was displayed on the right side while other information such as the path of .SEQ file, current sequence line being executed, its VI name, its specific test step and relating parameters were shown on the left. These slices of information were helpful when tracking measurement progress as well as when debugging measurement errors.

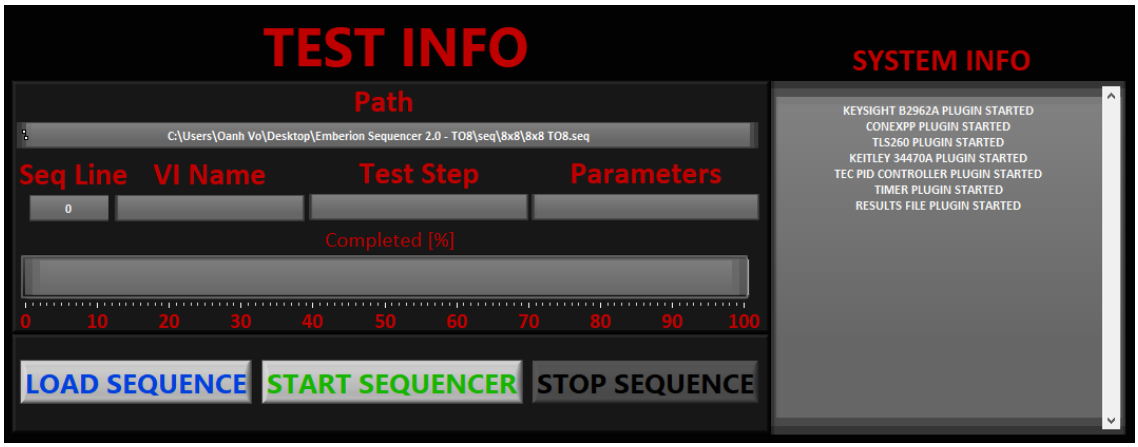


Figure 29. User interface of Emberion Sequencer.

A full measurement sequence lasted for about 1.5 hours. Data acquisition was fully achieved automatically by LabVIEW from the instruments, measurement PCBs and DUTs. Upon finishing the measurements, .CSV result files were ready for further data analysis. A measurement result file of a sample in .CSV format is illustrated in Appendix 4. In each .CSV result file of the temperature & power sweep, there were in total (13 temperature levels) x (32 ND-filter positions) x (50 read times) = 20800 data rows, plus one row reserved for the headers. Besides, there were 22 data columns, meaning each result file can be viewed as a 20801 x 22 data matrix. Eight DUTs were tested in total,

and there were only two out of eight devices functioned. This was likely due to the exceptional difficulties in wire bonding process of these DUTs.

GNU Octave 5.2.0 was the chosen software for data analysis. The goal was to find the temperature dependence of photodetector from the temperature & power sweep measurement from different perspectives. This was accomplished by plotting $V_{OFFSET(DARK)}$, $V_{OUT(LIGHT)}$, corresponding $V_{OUT\ (OFFSET\ CORRECTED)}$, ***Responsivity*** and ***Noise Equivalent Irradiance (NEI)***

1) *as a function of temperature* at a fixed irradiance (figures 30-32), and

2) *as a function of irradiance* at certain temperatures (figures 33-36).

A script was written to perform data analysis from these measurement result files. This full script can be found in Appendix 5. First, .CSV result files must be read, and measurement data of interest were extracted, such as output voltages during reset and exposure of the four pixels, temperatures, positions of ND-filter and corresponding extracted irradiances. The equations (19) to (23) were used to process measurement data:

$$V_{OUT(LIGHT)} = V_{RESET} - V_{EXPOSURE}\ (measured\ under\ light) \quad (19)$$

$$V_{OFFSET(DARK)} = V_{RESET} - V_{EXPOSURE}\ (measured\ under\ dark) \quad (20)$$

$$V_{OUT(OFFSET\ CORRECTED)} = V_{OUT(LIGHT)} - V_{OFFSET(DARK)}. \quad (21)$$

Output voltages at exposure and reset states were denoted as $V_{EXPOSURE}$ and V_{RESET}. $V_{OUT(LIGHT)}$ and $V_{OFFSET(DARK)}$ were the result voltages from light and dark measurement correspondingly. It was observed in the absence of light, the photo-detecting system still had some offset voltage. Therefore, understanding this non-zero output characteristic under dark was important as this directly affected the generated photovoltage and responsivity of the photodetectors (this will be further discussed in figure 31).

Knowing the general formula (1) for calculating responsivity of photodetectors presented in section 2.1.3, responsivity of the DUTs in this project was specifically determined as

$$Responsivity = \frac{V_{OUT(OFFSET\ CORRECTED)}}{Optical\ Power} = \frac{V_{OUT(LIGHT)} - V_{OFFSET(DARK)}}{Irradiance * Photosensitive\ area}. \quad (22)$$

Additionally, equation (7) defining Noise Equivalent Power (NEP = V_{noise} / Responsivity) in section 2.1.3 was also used to derive an important measure describing the performance of the DUTs: Noise Equivalent Irradiance, or NEI. This is the amount of irradiance needed to give the detector SNR of unity. It was given as

$$NEI = \frac{NEP}{Photosensitive\ area} = \frac{V_{noise}}{Responsivity * Photosensitive\ area}, \quad (23)$$

where responsivity was previously derived from equation (22), photosensitive area of DUTs was 20 µm by 20 µm, noise voltage level was the standard deviation of the offset voltage measured under dark conditions. The wavelength of light signal was fixed at 1550 nm, specific irradiance at this wavelength was extracted from the lamp calibration information. The unit of NEI is [W/m^2].

The results from a working device are illustrated in the next figures. It is also important to note that data presented are pre-amplification values.

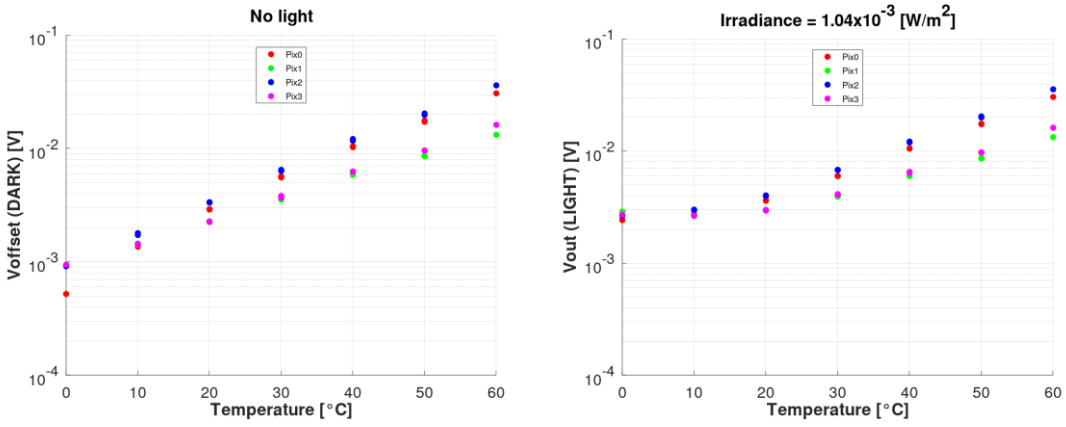


Figure 30. $V_{OFFSET(DARK)}$ when measured under dark conditions (left) and $V_{OUT(LIGHT)}$ when measured under light at irradiance of 1.04 x 10^{-3} W/m^2 (right).

From figure 30, it is shown in the left picture the plot of output voltages measured in the dark – $V_{OFFSET(DARK)}$ of four pixels in four different colours. This $V_{OFFSET(DARK)}$ was a non-zero offset voltage measured when there was no light coming to the detector. It can be observed that the higher the DUT temperature, the higher the offset voltage. The right

plot shows the output voltages measured under light of fixed irradiance 1.04 x 10^{-3} W/m^2. It had the similar trend: voltage increased when temperature increased. It appeared that at higher temperature, the magnitude of offset voltage under dark and under light were somewhat the same, meaning DUT's responsivity was not anymore high enough to generate signal over noise level.

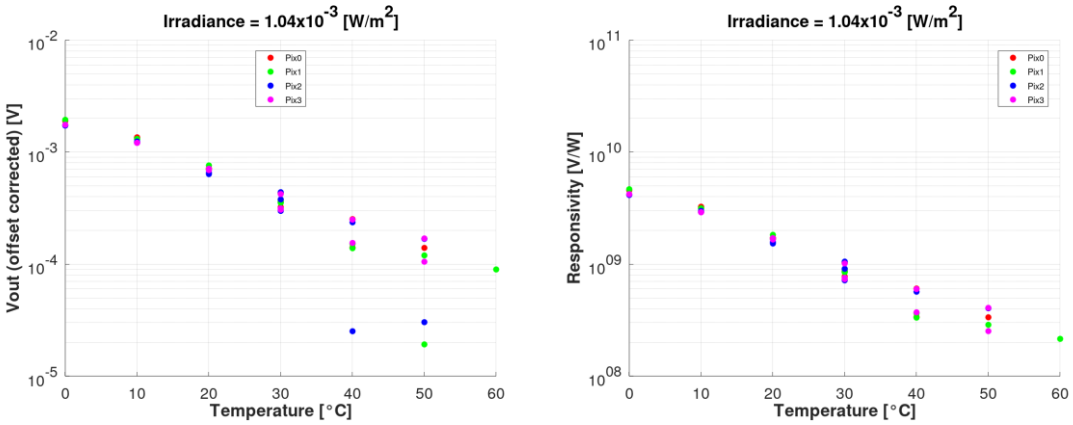


Figure 31. Corresponding $V_{OUT\ (OFFSET\ CORRECTED)}$ (left) and photo responsivity (right) of four pixels. The data was extracted at irradiance of 1.04 x 10^{-3} W/m^2.

Once the voltages measured under light and dark conditions were known, the actual photovoltage generated by the DUT – $V_{OUT\ (OFFSET\ CORRECTED)}$ was further calculated, and its plot is presented in figure 31 (left figure). It can be seen from figure 31 that, the higher the temperature, the lower the generated photovoltage. Furthermore, in the right plot of figure 31, the relationship of responsivity and temperature is depicted. For the same irradiance and photosensitive area, at lower temperature, the generated photovoltage was higher, hence better responsivity, and this responsivity lowered when temperature got higher. Moreover, it was also important to note that different pixels within a photodetector behaved slightly different from one another.

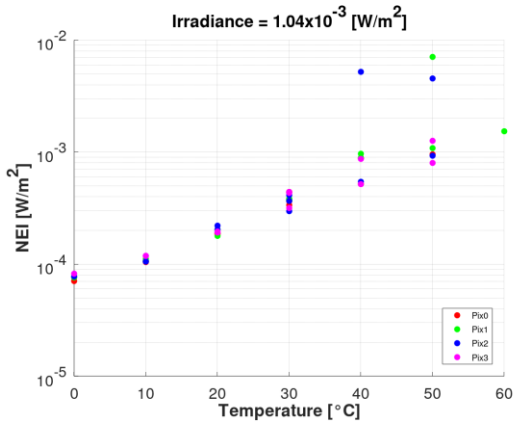


Figure 32. Corresponding NEI of four pixels as a function of temperature. The data was extracted at irradiance of 1.04 x 10^{-3} W/m^2.

Figure 32 describes the temperature dependence in terms of noise. At measured irradiance of 1.04 x 10^{-3} W/m^2, to achieve SNR of unity, the required irradiance falling on the detector was less at lower temperature, and vice versa. This means SNR was higher at low temperature and it started to decrease as temperature got higher. From about 40°C onwards, noise started to dominate the photovoltage signal.

The temperature dependence of photodetectors can also be demonstrated from a different perspective: plots of output voltages, responsivity and NEI measured at different temperatures as a function of irradiance (figures 33-36). These figures demonstrate the results when DUTs were under temperature & power sweep. Data in figures 33-36 was taken from the sweep of irradiance from 1.87 x 10^{-4} W/m^2 to 3.72 x 10^{-3} W/m^2, at different temperatures: 0°C, 30°C, and 60°C.

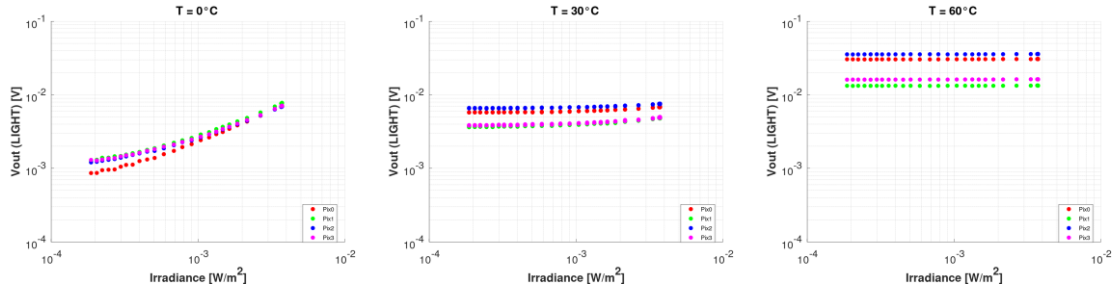


Figure 33. From left to right: $V_{OUT(LIGHT)}$ at different temperatures.

Figure 33 shows from left to right the output voltages $V_{OUT(LIGHT)}$ of the four pixels of DUT under the irradiance sweep at 0°C, 30°C, and 60°C respectively. It was shown at low temperatures (especially at 0°C) the expected proportional relationship between output voltage and irradiance level. However, at higher temperature (60°C), the sensitivity of DUT dropped, there seemed to be only noise, which as a result, gave the flat curves, meaning generated photovoltage was lower than noise.

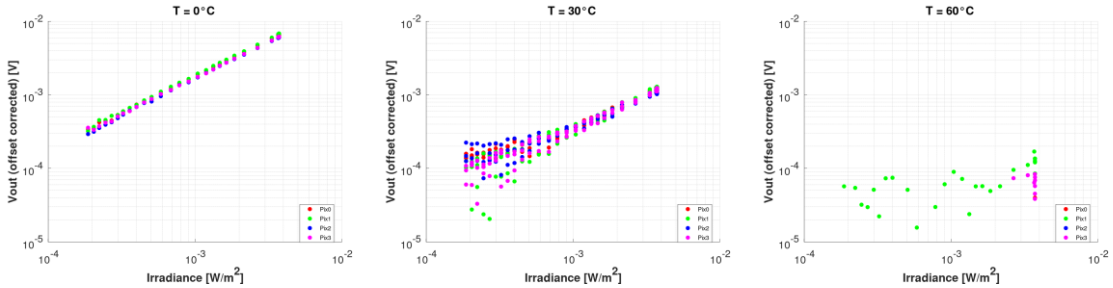


Figure 34. From left to right: $V_{OUT\ (OFFSET\ CORRECTED)}$ at different temperatures.

Once offset voltage observed under dark condition was removed from the output voltage measured under light, the residual voltage $V_{OUT(OFFSET\ CORRECTED)}$, meaning generated photovoltage was plotted as a function of irradiance at 0°C, 30°C, and 60°C, as shown in figure 34. A similar trend was monitored: photovoltage increased as irradiance got higher when measured at lower temperatures (0°C, 30°C). At higher temperature (60°C), the photodetector seemed to become insensitive and noise was dominating.

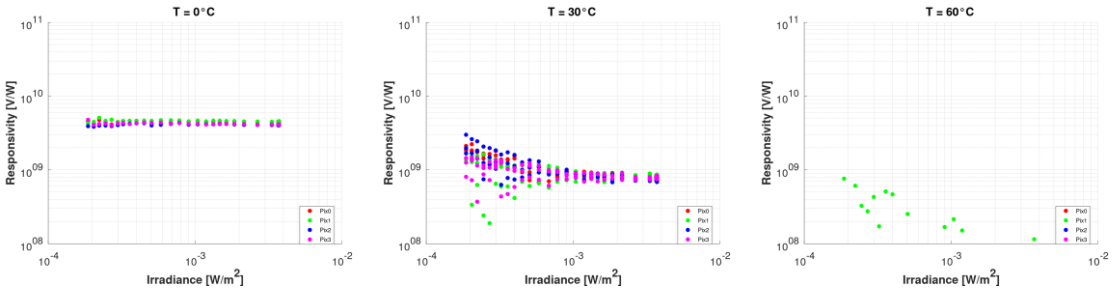


Figure 35. From left to right: Responsivity at different temperatures.

The temperature dependence of photodetector was further demonstrated in terms of responsivity in figure 35. Based on the known generated photovoltage, the corresponding responsivity was plotted at different temperatures as a function of irradiance. The

expected responsivity vs. irradiance plot could be easily seen at lower temperatures (0°C, 30°C). However, once temperature was higher (60°C), the responsivity of photodetector dropped dramatically, most of the pixels were no longer sensitive, only noise was seen.

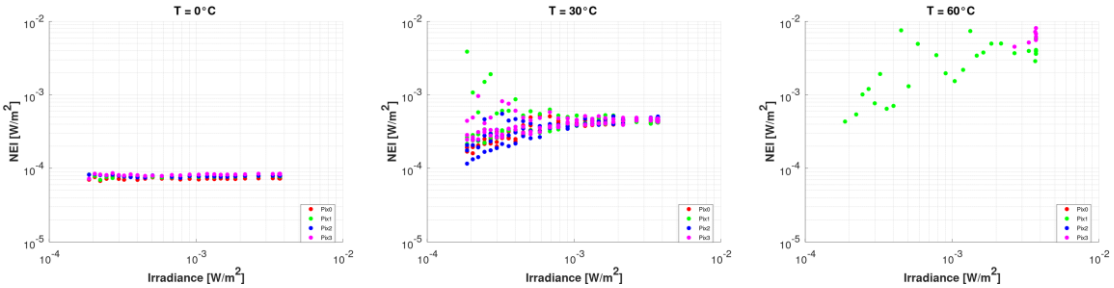


Figure 36. From left to right: NEI at different temperatures.

From the NEI point of view, figure 36 shows the required irradiance to yield SNR of one was lower at 0°C, and it needed more irradiance to produce such SNR as the temperature got higher. At 60°C, noise was had a major influence, the photodetector was no longer sensitive.

6 Discussions and Conclusions

The development of a temperature measurement system including hardware, software as well as the actual temperature measurement execution and data analysis were initially set to be the main goals of this thesis project. It can be concluded that this set of goals was successfully achieved. In terms of hardware, a measurement PCB was designed using PADS, then it was manufactured, and tested to be functional. From software point of view, modifications of FPGA Pixel Sequencer for desired pixel switching was accomplished; additional LabVIEW code was amended, properly run, and gave expected outputs; test sequence file for measurement automation was built and worked as intended. Furthermore, the temperature measurements can be said to be conducted in a proper manner. Dark-signals were measured along side with light-signals at different temperature levels, which as a result, gave a good estimation of sensor's (1) offset in dark, (2) response to light and (3) noise level present in the detection system as a function of temperature. Number of data samples taken was sufficiently high, which helped

maintaining the accuracy of measured data. Finally, with the help of GNU Octave, measurements results were efficiently analysed, and the critical plots were generated for better data visualization, thus important propositions or findings could be immediately concluded.

Even though only two out of eight tested Emberion R&D photodetectors were working, which was likely due to high complexity in packaging process, it was enough to study the temperature dependence of these photodetectors. The temperature dependence which was mainly due to thermally generated charge carriers was observed, and this outcome provides important insights for future developments of high-performance photodetectors of Emberion. This strong dependence of photodetectors on temperature also confirms the challenges that photodetector technology in general encounters: developing uncooled photodetectors means performance must suffer, while photodetectors operating with coolers would increase cost and power consumption. This suggests that further studies and experiments are needed in this area in order to improve the temperature characteristics of photodetectors.

References

1 Emberion [Internet]. Emberion Company. 2020. Accessed 27 April 2020. https://www.emberion.com/company/

2 Ready J. Fundamentals of Photonics – Module 1.6: Optical Detectors and Human Vision [e-publication]. SPIE. 2020. Accessed 12 February 2020. https://spie.org/publications/fundamentals-of-photonics-modules

3 Lerner E. Introduction to photodetectors and applications [e-article]. Laser Focus World. 2000. Accessed 25 March 2020. https://www.laserfocusworld.com/detectors-imaging/article/16555513/introduction-to-photodetectors-and-applications

4 Omnes F. Introduction to Semiconductor Photodetectors. 2009.

5 Vincent J, Hodges S, Vampola J, Stegall M, Pierce G. Fundamentals of infrared and visible detector operation and testing [e-book]. 2nd ed. Hoboken: John Wiley & Sons; 2015. Accessed 30 January 2020. https://ebookcentral.proquest.com/lib/metropoliaebooks/detail.action?docID=4436069#

6 OSI Optoelectronics. Photovoltaic Devices [Internet]. 2013. Accessed 3 April 2020. http://www.osioptoelectronics.com/standard-products/silicon-photodiodes/general-purpose-photodiodes/photovoltaic-devices.aspx

7 Konstantatos G, Sargent EH, editors. Colloidal quantum dot optoelectronics and photovoltaics. Cambridge: Cambridge University Press; 2013.

8 Sze S, Ng K. Physics of semiconductor devices. 3rd ed. New Jersey: John Wiley & Sons; 2007.

9 Kularatna N. Electronic circuit design: from concept to implementation. Boca Raton: CRC Press; 2008.

10 Lutz G. Semiconductor radiation detectors. 1st ed. Berlin: Springer; 2007.

11 Kogan S. Electronic noise and fluctuations in solids. Cambridge: Cambridge University Press; 2008.

12 Rogalski A. Infrared detectors. 2nd ed. Boca Raton: CRC Press; 2011.

13 Orton J. The story of semiconductors. Oxford: Oxford University Press; 2004.

14 OSI Optoelectronics. Photodiode Characteristics and Applications [Internet]. 2013. Accessed 3 April 2020. http://www.osioptoelectronics.com/application-notes/AN-Photodiode-Parameters-and-Characteristics.pdf

15 Hamamatsu. Si photodiodes [Internet]. Accessed 3 April 2020. https://www.hamamatsu.com/resources/pdf/ssd/e02_handbook_si_photodiode.pdf

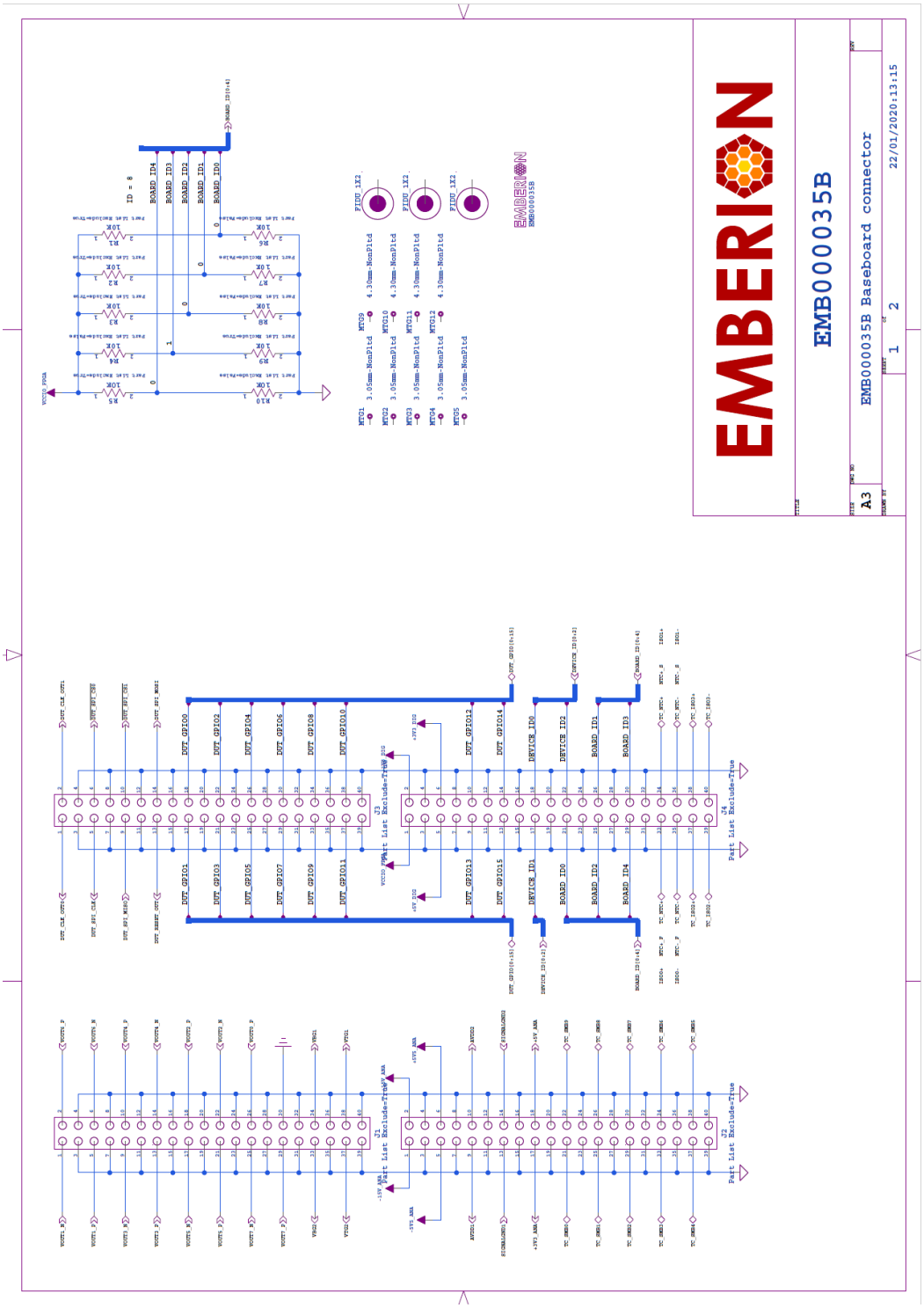
16 Grundmann M. The physics of semiconductors. Berlin: Springer. 2006.

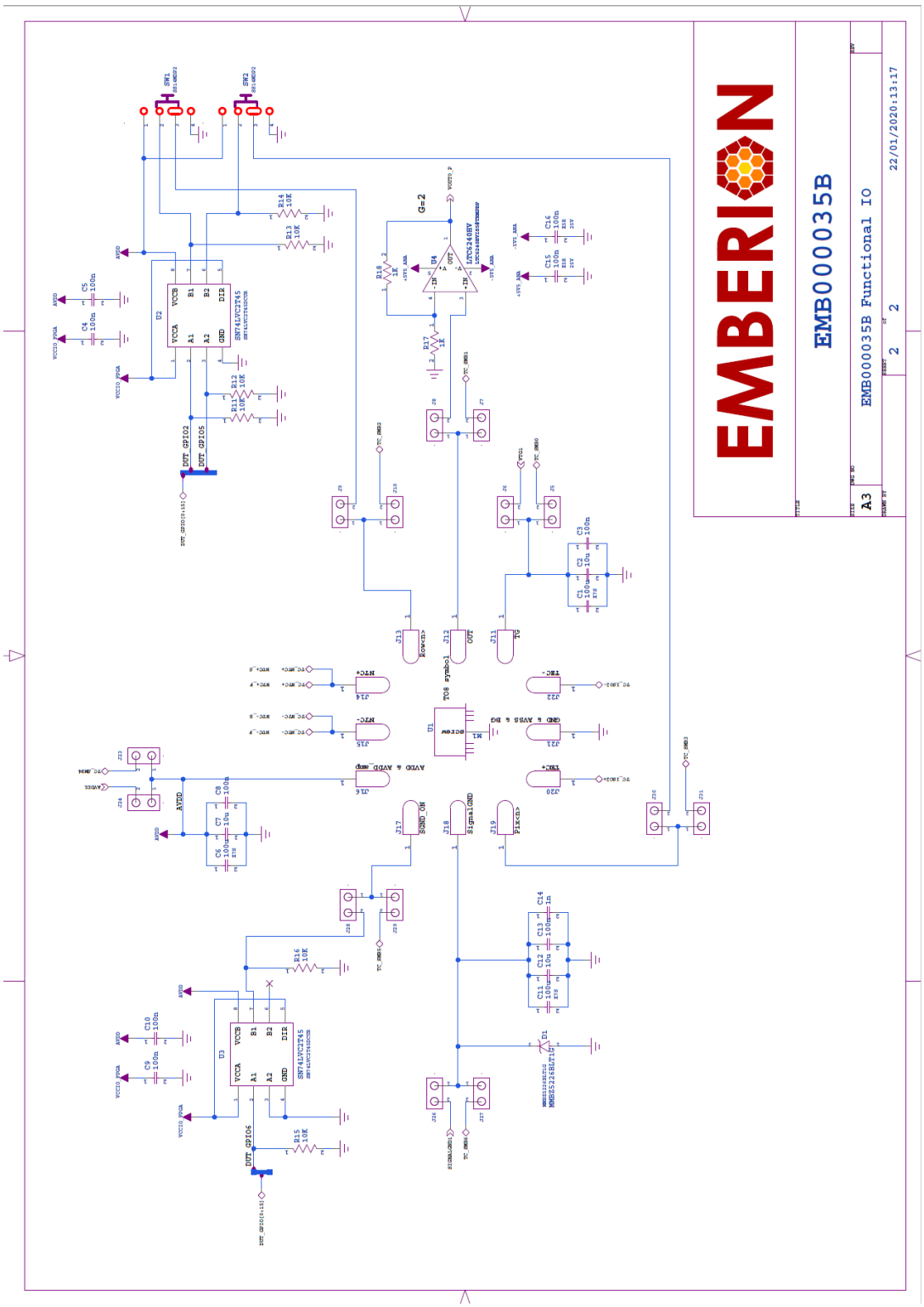
17 Leach W.M. 2020. Accessed 3 May 2020. http://leachlegacy.ece.gatech.edu/ece3040/notes/chap02.pdf

18 Arora N. Mosfet modeling for VLSI simulation – Theory and Practice. Singapore: World Scientific. 2007.

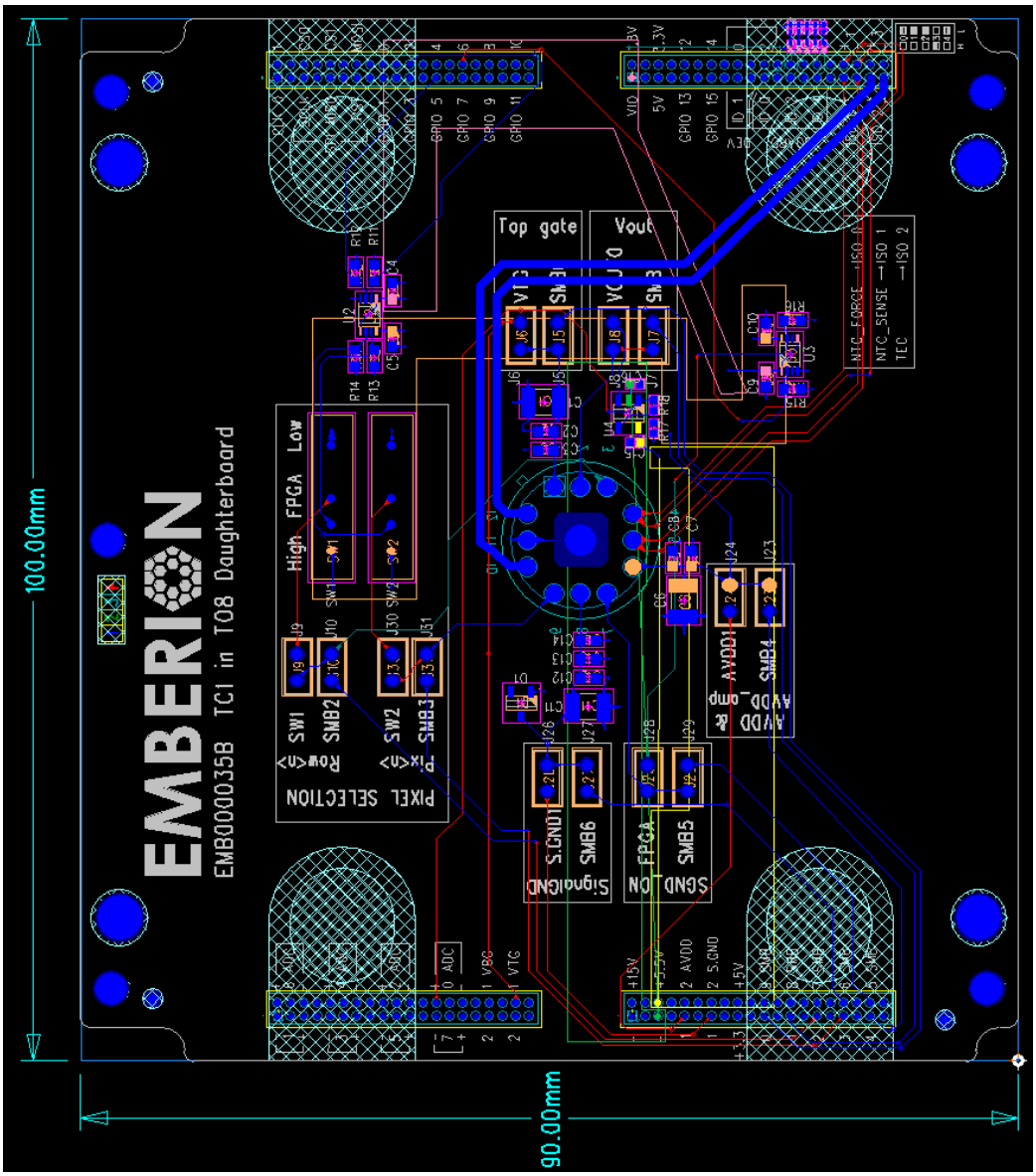
19 Piotrowski J, Rogalski A. High-operating-temperature infrared photodetectors. Bellingham, Wash.: SPIE. 2007.

Appendix 1: Daughterboard Schematic




EMBERION
EMB000035B
EMB000035B Functional IO
A3
22/01/2020 13:17
G=2

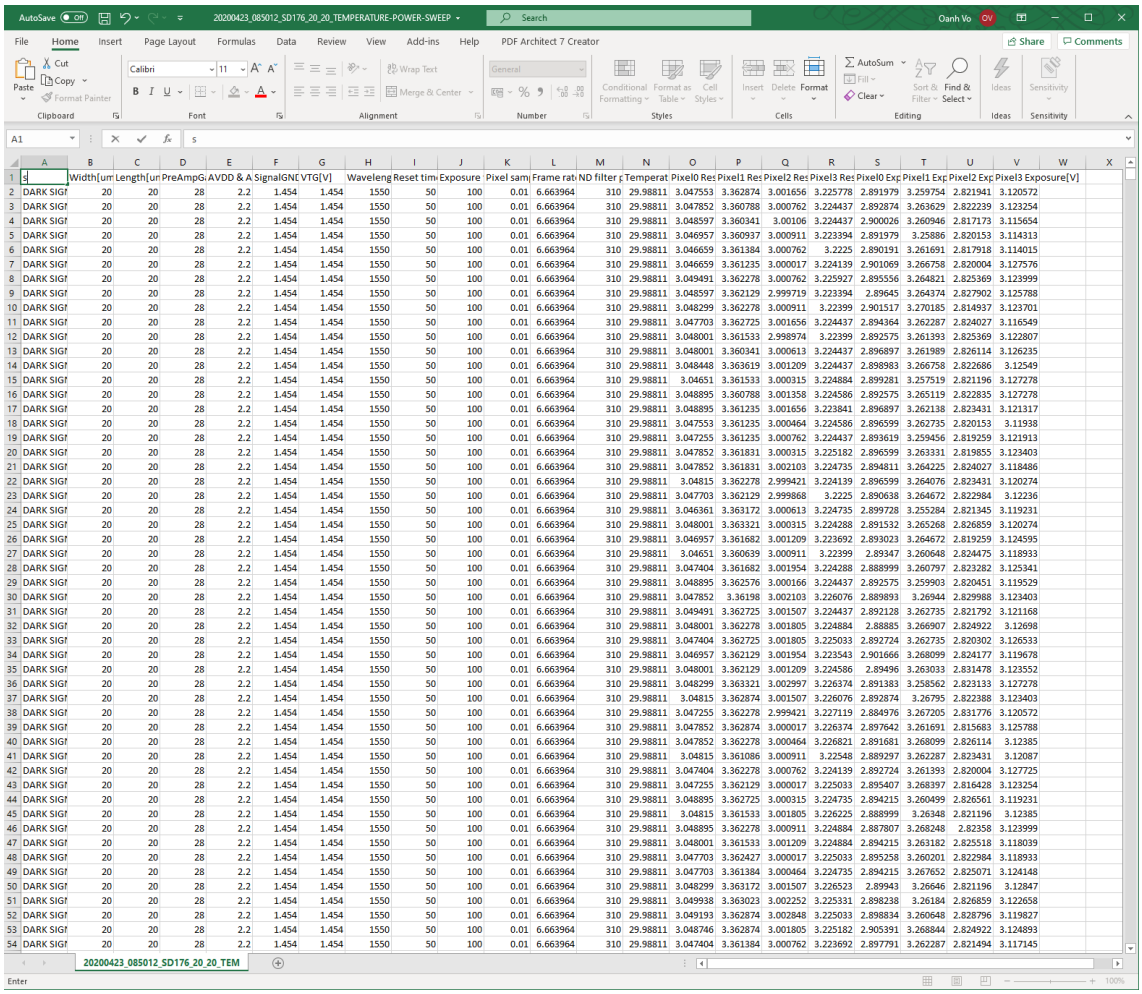
Appendix 2: Daughterboard Layout



Appendix 3: Test Sequence: Temperature & Power Sweep

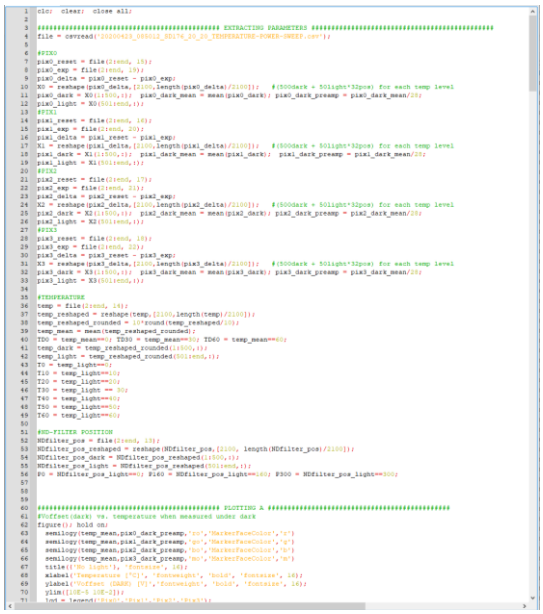
```
120
121 //################ START TEMPERATURE & POWER SWEEP 1550NM ###############################################################
122 Results file.vi     create.file data(pwbId,SampleId,width,length) name(Temperature-Power-Sweep) data2() name2()
123 Results file.vi     create.headers.for.setup setup(8X8 TO8)
124 TLS260.vi           set.wavelength wavelength(1550) autoGrating(yes) autoFilter(yes)
125
126 //############### SWEEP FROM 0 TO 60 CELSIUS ############################################
127 sweep1 targetT(30,10,60)
128
129 TEC PID Controller.vi   start.temperature.control targetT()
130
131 sweep2 position(0,10,310)
132
133 ConexPP.vi          move.absolute position()
134 8x8.vi              read.data frames(50)
135 Results file.vi     save.batch.data comment() setup(8X8) batchDataName(8x8)
    data(width,length,preAmpGain,avdd1,signalgnd1,vtg1,wavelength,resetTime,exposureTime,pixelSampleTime,frameRate,position,
    temperature,8x8)
136
137 end sweep2
138
139 end sweep1
140
141 sweep1 targetT(50,-10,0)
142
143 TEC PID Controller.vi   start.temperature.control targetT()
144
145 sweep2 position(0,10,310)
146
147 ConexPP.vi          move.absolute position()
148 8x8.vi              read.data frames(50)
149 Results file.vi     save.batch.data comment() setup(8X8) batchDataName(8x8)
    data(width,length,preAmpGain,avdd1,signalgnd1,vtg1,wavelength,resetTime,exposureTime,pixelSampleTime,frameRate,position,
    temperature,8x8)
150
151 end sweep2
152
153 end sweep1
154
155 sweep1 targetT(10,10,30)
156
157 TEC PID Controller.vi   start.temperature.control targetT()
158
159 sweep2 position(0,10,310)
160
161 ConexPP.vi          move.absolute position()
162 8x8.vi              read.data frames(50)
163 Results file.vi     save.batch.data comment() setup(8X8) batchDataName(8x8)
    data(width,length,preAmpGain,avdd1,signalgnd1,vtg1,wavelength,resetTime,exposureTime,pixelSampleTime,frameRate,position,
    temperature,8x8)
164
165 end sweep2
166
167 end sweep1
168
169 TEC PID Controller.vi   stop.temperature.control
170 //################ END OF TEMPERATURE & POWER SWEEP ###################################################################
```

Appendix 4: A .CSV Measurement Result File

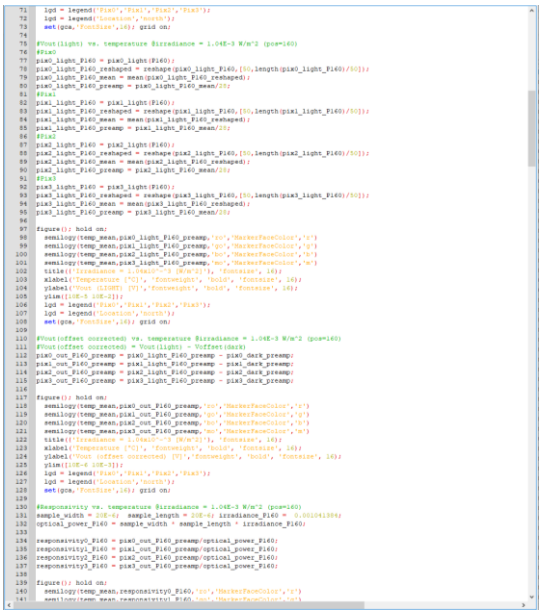


s	Width[um	Length[un	PreAmpGa	AVDD & A	SignalGNE	VTG[V]	Waveleng	Reset tim	Exposure	Pixel sam	Frame rat	ND filter	Temperat	Pixel0 Res	Pixel1 Res	Pixel2 Res	Pixel3 Res	Pixel0 Exp	Pixel1 Exp	Pixel2 Exp	Pixel3 Exposure[V]
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.047553	3.362874	3.001656	3.225778	2.891979	3.259754	2.821941	3.120572
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.047852	3.360788	3.000762	3.224437	2.892874	3.263629	2.822239	3.123254
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.048597	3.360341	3.00106	3.224437	2.900026	3.260946	2.817173	3.115654
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.046957	3.360937	3.000911	3.223394	2.891979	3.25886	2.820153	3.114313
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.046659	3.361384	3.000762	3.2225	2.890191	3.261691	2.817918	3.114015
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.046659	3.361235	3.000017	3.224139	2.901069	3.266758	2.820004	3.127576
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.049491	3.362278	3.000762	3.225927	2.895556	3.264821	2.825369	3.123999
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.048597	3.362129	2.999719	3.223394	2.89645	3.264374	2.827902	3.125788
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.048299	3.362278	3.000911	3.22399	2.901517	3.270185	2.814937	3.123701
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.047703	3.362725	3.001656	3.224437	2.894364	3.262287	2.824027	3.116549
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.048001	3.361533	2.998974	3.22399	2.892575	3.261393	2.825369	3.122807
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.048001	3.360341	3.000613	3.224437	2.896897	3.261989	2.826114	3.126235
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.048448	3.363619	3.001209	3.224437	2.898983	3.266758	2.822686	3.12549
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.04651	3.361533	3.000315	3.224884	2.899281	3.257519	2.821196	3.127278
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.048895	3.360788	3.001358	3.224586	2.892575	3.265119	2.822835	3.127278
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.048895	3.361235	3.001656	3.223841	2.896897	3.262138	2.823431	3.121317
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.047553	3.361235	3.000464	3.224586	2.896599	3.262735	2.820153	3.11938
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.047255	3.361235	3.000762	3.224437	2.893619	3.259456	2.819259	3.121913
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.047852	3.361831	3.000315	3.225182	2.896599	3.263331	2.819855	3.123403
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.047852	3.361831	3.002103	3.224735	2.894811	3.264225	2.824027	3.118486
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.04815	3.362278	2.999421	3.224139	2.896599	3.264076	2.823431	3.120274
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.047703	3.362129	2.999868	3.2225	2.890638	3.264672	2.822984	3.12236
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.046361	3.363172	3.000613	3.224735	2.899728	3.255284	2.821345	3.119231
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.048001	3.363321	3.000315	3.224288	2.891532	3.265268	2.826859	3.120274
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.046957	3.361682	3.001209	3.223692	2.893023	3.264672	2.819259	3.124595
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.04651	3.360639	3.000911	3.22399	2.89347	3.260648	2.824475	3.118933
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.047404	3.361682	3.001954	3.224288	2.888999	3.260797	2.823282	3.125341
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.048895	3.362576	3.000166	3.224437	2.892575	3.259903	2.820451	3.119529
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.047852	3.36198	3.002103	3.226076	2.889893	3.26944	2.829988	3.123403
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.049491	3.362725	3.001507	3.224437	2.892128	3.262735	2.821792	3.121168
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.048001	3.362278	3.001805	3.224884	2.88885	3.266907	2.824922	3.12698
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.047404	3.362725	3.001805	3.225033	2.892724	3.262735	2.820302	3.126533
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.046957	3.362129	3.001954	3.223543	2.901666	3.268099	2.824177	3.119678
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.048001	3.362129	3.001209	3.224586	2.89496	3.263033	2.831478	3.123552
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.048299	3.363321	3.002997	3.226374	2.891383	3.258562	2.823133	3.127278
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.04815	3.362874	3.001507	3.226076	2.892874	3.26795	2.822388	3.123403
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.047255	3.362278	2.999421	3.227119	2.884976	3.267205	2.831776	3.120572
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.047852	3.362874	3.000017	3.226374	2.897642	3.261691	2.815683	3.125788
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.047852	3.362278	3.000464	3.226821	2.891681	3.268099	2.826114	3.12385
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.04815	3.361086	3.000911	3.22548	2.889297	3.262287	2.823431	3.12087
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.047404	3.362278	3.000762	3.224139	2.892724	3.261393	2.820004	3.127725
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.047255	3.362129	3.000017	3.225033	2.895407	3.268397	2.816428	3.123254
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.048895	3.362725	3.000315	3.224735	2.894215	3.260499	2.826561	3.119231
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.04815	3.361533	3.001805	3.226225	2.888999	3.26348	2.821196	3.12385
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.048895	3.362278	3.000911	3.224884	2.887807	3.268248	2.82358	3.123999
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.048001	3.361533	3.001209	3.224884	2.894215	3.263182	2.825518	3.118039
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.047703	3.362427	3.000017	3.225033	2.895258	3.260201	2.822984	3.118933
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.047703	3.361384	3.000464	3.224735	2.894215	3.267652	2.825071	3.124148
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.048299	3.363172	3.001507	3.226523	2.89943	3.26646	2.821196	3.12847
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.049938	3.363023	3.002252	3.225331	2.898238	3.26184	2.826859	3.122658
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.049193	3.362874	3.002848	3.225033	2.898834	3.260648	2.828796	3.119827
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.048746	3.362874	3.001805	3.225182	2.905391	3.268844	2.824922	3.124893
DARK SIGI	20	20	28	2.2	1.454	1.454	1550	50	100	0.01	6.663964	310	29.98811	3.047404	3.361384	3.000762	3.223692	2.897791	3.262287	2.821494	3.117145

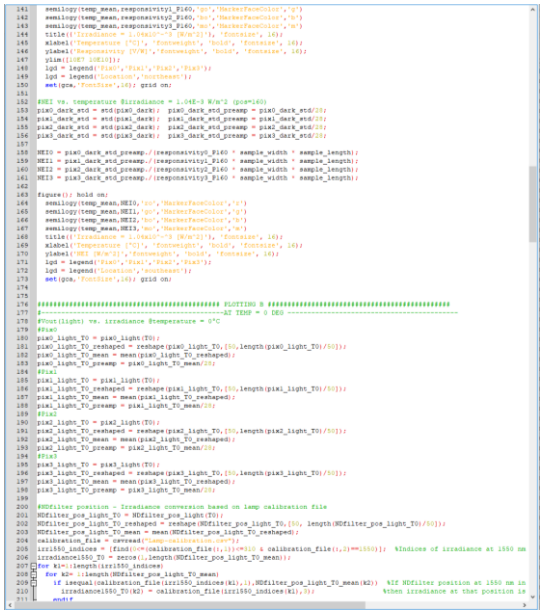
Appendix 5: GNU Octave Script for Data Analysis



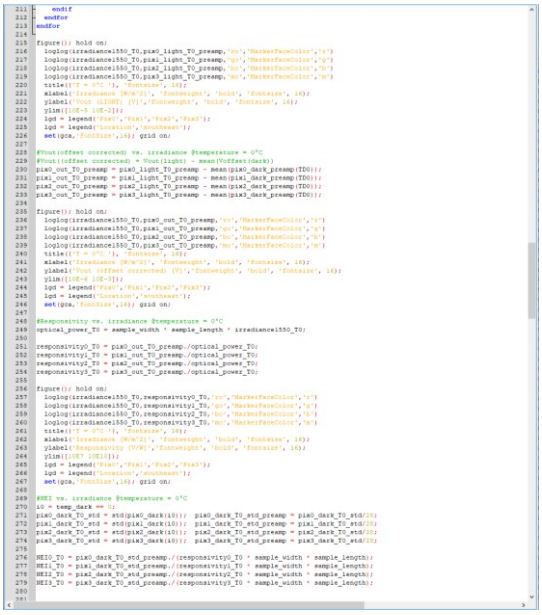
```
clc  clear;  close all;

########################################## EXTRACTING PARAMETERS ##########################################
file = cstread('D1200x63_600x12_EN578_20_DC_TEMPERATURE-POWER-SWEEP.csv');

#PIN0
pin0_reset = file(2:end, 15);
pin0_exp = file(2:end, 16);
pin0_delta = pin0_reset - pin0_exp;
X0 = reshape(pin0_delta,[2100,length(pin0_delta)/2100]);   #(500dark + 50light*32pos) for each temp level
pin0_dark = X0(1:1500,:);  pin0_dark_mean = mean(pin0_dark); pin0_dark_preamp = pin0_dark_mean/20;
pin0_light = X0(501:end,:);
#PIN1
pin1_reset = file(2:end, 16);
pin1_exp = file(2:end, 20);
pin1_delta = pin1_reset - pin1_exp;
X1 = reshape(pin1_delta,[2100,length(pin1_delta)/2100]);   #(500dark + 50light*32pos) for each temp level
pin1_dark = X1(1:1500,:);  pin1_dark_mean = mean(pin1_dark);  pin1_dark_preamp = pin1_dark_mean/20;
pin1_light = X1(501:end,:);
#PIN2
pin2_reset = file(2:end, 17);
pin2_exp = file(2:end, 21);
pin2_delta = pin2_reset - pin2_exp;
X2 = reshape(pin2_delta,[2100,length(pin2_delta)/2100]);   #(500dark + 50light*32pos) for each temp level
pin2_dark = X2(1:1500,:);  pin2_dark_mean = mean(pin2_dark); pin2_dark_preamp = pin2_dark_mean/20;
pin2_light = X2(501:end,:);
#PIN3
pin3_reset = file(2:end, 18);
pin3_exp = file(2:end, 22);
pin3_delta = pin3_reset - pin3_exp;
X3 = reshape(pin3_delta,[2100,length(pin3_delta)/2100]);   #(500dark + 50light*32pos) for each temp level
pin3_dark = X3(1:1500,:);  pin3_dark_mean = mean(pin3_dark); pin3_dark_preamp = pin3_dark_mean/20;
pin3_light = X3(501:end,:);

#TEMPERATURE
temp = file(2:end, 14);
temp_reshaped = reshape(temp,[2100,length(temp)/2100]);
temp_reshaped_rounded = 10*round(temp_reshaped/10);
temp_mean = mean(temp_reshaped_rounded);
TD0 = temp_mean==0; TD30 = temp_mean==30; TD60 = temp_mean==60;
temp_dark = temp_reshaped_rounded(1:1500,:);
temp_light = temp_reshaped_rounded(501:end,:);
T0 = temp_light==0;
T10 = temp_light==10;
T20 = temp_light==20;
T30 = temp_light == 30;
T40 = temp_light==40;
T50 = temp_light==50;
T60 = temp_light==60;

#ND-FILTER POSITION
NDfilter_pos = file(2:end, 13);
NDfilter_pos_reshaped = reshape(NDfilter_pos,[2100, length(NDfilter_pos)/2100]);
NDfilter_pos_dark = NDfilter_pos_reshaped(1:1500,:);
NDfilter_pos_light = NDfilter_pos_reshaped(501:end,:);
P0 = NDfilter_pos_light==0; P160 = NDfilter_pos_light==160; P300 = NDfilter_pos_light==300;


########################################## PLOTTING A ##########################################
#Voffset(dark) vs. temperature when measured under dark
figure(1); hold on;
 semilogy(temp_mean,pin0_dark_preamp,'ro','MarkerFaceColor','r')
 semilogy(temp_mean,pin1_dark_preamp,'go','MarkerFaceColor','g')
 semilogy(temp_mean,pin2_dark_preamp,'bo','MarkerFaceColor','b')
 semilogy(temp_mean,pin3_dark_preamp,'mo','MarkerFaceColor','m')
 title({'NO light'}, 'fontsize', 20);
 xlabel('Temperature (°C)', 'fontweight', 'bold', 'fontsize', 16);
 ylabel('Voffset (DARK) [V]','fontweight', 'bold', 'fontsize', 16);
 ylim([10E-5 10E-2]);
 lgd = legend('Pin0','Pin1','Pin2','Pin3');
```



```
 lgd = legend('Pin0','Pin1','Pin2','Pin3');
 lgd = legend('Location','north');
 set(gca,'FontSize',16); grid on;

#Vout(light) vs. temperature @irradiance = 1.04E-3 W/m^2 (pos=160)
#Pin0
pin0_light_P160 = pin0_light(P160);
pin0_light_P160_reshaped = reshape(pin0_light_P160,[50,length(pin0_light_P160)/50]);
pin0_light_P160_mean = mean(pin0_light_P160_reshaped);
pin0_light_P160_preamp = pin0_light_P160_mean/20;
#Pin1
pin1_light_P160 = pin1_light(P160);
pin1_light_P160_reshaped = reshape(pin1_light_P160,[50,length(pin1_light_P160)/50]);
pin1_light_P160_mean = mean(pin1_light_P160_reshaped);
pin1_light_P160_preamp = pin1_light_P160_mean/20;
#Pin2
pin2_light_P160 = pin2_light(P160);
pin2_light_P160_reshaped = reshape(pin2_light_P160,[50,length(pin2_light_P160)/50]);
pin2_light_P160_mean = mean(pin2_light_P160_reshaped);
pin2_light_P160_preamp = pin2_light_P160_mean/20;
#Pin3
pin3_light_P160 = pin3_light(P160);
pin3_light_P160_reshaped = reshape(pin3_light_P160,[50,length(pin3_light_P160)/50]);
pin3_light_P160_mean = mean(pin3_light_P160_reshaped);
pin3_light_P160_preamp = pin3_light_P160_mean/20;

figure(2); hold on;
 semilogy(temp_mean,pin0_light_P160_preamp,'ro','MarkerFaceColor','r')
 semilogy(temp_mean,pin1_light_P160_preamp,'go','MarkerFaceColor','g')
 semilogy(temp_mean,pin2_light_P160_preamp,'bo','MarkerFaceColor','b')
 semilogy(temp_mean,pin3_light_P160_preamp,'mo','MarkerFaceColor','m')
 title({'Irradiance = 1.04x10^-3 (W/m^2)'}, 'fontsize', 16);
 xlabel('Temperature (°C)', 'fontweight', 'bold', 'fontsize', 16);
 ylabel('Vout (LIGHT) [V]','fontweight', 'bold', 'fontsize', 16);
 ylim([10E-5 10E-2]);
 lgd = legend('Pin0','Pin1','Pin2','Pin3');
 lgd = legend('Location','north');
 set(gca,'FontSize',16); grid on;

#Vout(offset corrected) vs. temperature @irradiance = 1.04E-3 W/m^2 (pos=160)
#Vout(offset corrected) = Vout(light) - Voffset(dark)
pin0_out_P160_preamp = pin0_light_P160_preamp - pin0_dark_preamp;
pin1_out_P160_preamp = pin1_light_P160_preamp - pin1_dark_preamp;
pin2_out_P160_preamp = pin2_light_P160_preamp - pin2_dark_preamp;
pin3_out_P160_preamp = pin3_light_P160_preamp - pin3_dark_preamp;

figure(3); hold on;
 semilogy(temp_mean,pin0_out_P160_preamp,'ro','MarkerFaceColor','r')
 semilogy(temp_mean,pin1_out_P160_preamp,'go','MarkerFaceColor','g')
 semilogy(temp_mean,pin2_out_P160_preamp,'bo','MarkerFaceColor','b')
 semilogy(temp_mean,pin3_out_P160_preamp,'mo','MarkerFaceColor','m')
 title({'Irradiance = 1.04x10^-3 (W/m^2)'}, 'fontsize', 16);
 xlabel('Temperature (°C)', 'fontweight', 'bold', 'fontsize', 16);
 ylabel('Vout (offset corrected) [V]','fontweight', 'bold', 'fontsize', 16);
 ylim([10E-6 10E-3]);
 lgd = legend('Pin0','Pin1','Pin2','Pin3');
 lgd = legend('Location','north');
 set(gca,'FontSize',16); grid on;

#Responsivity vs. temperature @irradiance = 1.04E-3 W/m^2 (pos=160)
sample_width = 20E-6;  sample_length = 20E-6; irradiance_P160 = 1.001041384;
optical_power_P160 = sample_width * sample_length * irradiance_P160;

responsivity0_P160 = pin0_out_P160_preamp/optical_power_P160;
responsivity1_P160 = pin1_out_P160_preamp/optical_power_P160;
responsivity2_P160 = pin2_out_P160_preamp/optical_power_P160;
responsivity3_P160 = pin3_out_P160_preamp/optical_power_P160;

figure(4); hold on;
 semilogy(temp_mean,responsivity0_P160,'ro','MarkerFaceColor','r')
```



```
 semilogy(temp_mean,responsivity1_P160,'go','MarkerFaceColor','g')
 semilogy(temp_mean,responsivity2_P160,'bo','MarkerFaceColor','b')
 semilogy(temp_mean,responsivity3_P160,'mo','MarkerFaceColor','m')
 title({'Irradiance = 1.04x10^-3 (W/m^2)'}, 'fontsize', 16);
 xlabel('Temperature (°C)', 'fontweight', 'bold', 'fontsize', 16);
 ylabel('Responsivity (V/W)','fontweight', 'bold', 'fontsize', 16);
 ylim([10E7 10E10]);
 lgd = legend('Pin0','Pin1','Pin2','Pin3');
 lgd = legend('Location','southeast');
 set(gca,'FontSize',16); grid on;

#SEI vs. temperature @irradiance = 1.04E-3 W/m^2 (pos=160)
pin0_dark_std = std(pin0_dark);  pin0_dark_std_preamp = pin0_dark_std/20;
pin1_dark_std = std(pin1_dark);  pin1_dark_std_preamp = pin1_dark_std/20;
pin2_dark_std = std(pin2_dark);  pin2_dark_std_preamp = pin2_dark_std/20;
pin3_dark_std = std(pin3_dark);  pin3_dark_std_preamp = pin3_dark_std/20;

SEI0 = pin0_dark_std_preamp./(responsivity0_P160 * sample_width * sample_length);
SEI1 = pin1_dark_std_preamp./(responsivity1_P160 * sample_width * sample_length);
SEI2 = pin2_dark_std_preamp./(responsivity2_P160 * sample_width * sample_length);
SEI3 = pin3_dark_std_preamp./(responsivity3_P160 * sample_width * sample_length);

figure(5); hold on;
 semilogy(temp_mean,SEI0,'ro','MarkerFaceColor','r')
 semilogy(temp_mean,SEI1,'go','MarkerFaceColor','g')
 semilogy(temp_mean,SEI2,'bo','MarkerFaceColor','b')
 semilogy(temp_mean,SEI3,'mo','MarkerFaceColor','m')
 title({'Irradiance = 1.04x10^-3 (W/m^2)'}, 'fontsize', 16);
 xlabel('Temperature (°C)', 'fontweight', 'bold', 'fontsize', 16);
 ylabel('SEI (W/m^2)','fontweight', 'bold', 'fontsize', 16);
 lgd = legend('Pin0','Pin1','Pin2','Pin3');
 lgd = legend('location','southeast');
 set(gca,'FontSize',16); grid on;


########################################## PLOTTING B ##########################################
#-------------------------------------------AT TEMP = 0 DEG ------------------------------------------
#Vout(light) vs. irradiance @temperature = 0°C
#Pin0
pin0_light_T0 = pin0_light(T0);
pin0_light_T0_reshaped = reshape(pin0_light_T0,[50,length(pin0_light_T0)/50]);
pin0_light_T0_mean = mean(pin0_light_T0_reshaped);
pin0_light_T0_preamp = pin0_light_T0_mean/20;
#Pin1
pin1_light_T0 = pin1_light(T0);
pin1_light_T0_reshaped = reshape(pin1_light_T0,[50,length(pin1_light_T0)/50]);
pin1_light_T0_mean = mean(pin1_light_T0_reshaped);
pin1_light_T0_preamp = pin1_light_T0_mean/20;
#Pin2
pin2_light_T0 = pin2_light(T0);
pin2_light_T0_reshaped = reshape(pin2_light_T0,[50,length(pin2_light_T0)/50]);
pin2_light_T0_mean = mean(pin2_light_T0_reshaped);
pin2_light_T0_preamp = pin2_light_T0_mean/20;
#Pin3
pin3_light_T0 = pin3_light(T0);
pin3_light_T0_reshaped = reshape(pin3_light_T0,[50,length(pin3_light_T0)/50]);
pin3_light_T0_mean = mean(pin3_light_T0_reshaped);
pin3_light_T0_preamp = pin3_light_T0_mean/20;

#NDfilter position - Irradiance conversion based on lamp calibration file
NDfilter_pos_light_T0 = NDfilter_pos_light(T0);
NDfilter_pos_light_T0_reshaped = reshape(NDfilter_pos_light_T0,[50, length(NDfilter_pos_light_T0)/50]);
NDfilter_pos_light_T0_mean = mean(NDfilter_pos_light_T0_reshaped);
calibration_file = csvread('Lamp-calibration.csv');
irr1550_indices = [find(0<=(calibration_file(:,1)-0)<10 & calibration_file(:,2)==1550)];  %Indices of irradiance at 1550 nm
irradiance1550_T0 = zeros(1,length(NDfilter_pos_light_T0_mean));
for k1=1:length(irr1550_indices)
  for k2= 1:length(NDfilter_pos_light_T0_mean)
    if isequal(calibration_file(irr1550_indices(k1),1),NDfilter_pos_light_T0_mean(k2))  %IF NDfilter position at 1550 nm is
      irradiance1550_T0(k2) = calibration_file(irr1550_indices(k1),3);                  %then irradiance at that position is
    endif
```



```
  endif
 endfor
endfor

figure(1); hold on;
 loglog(irradiance1550_T0,pin0_light_T0_preamp,'ro','MarkerFaceColor','r')
 loglog(irradiance1550_T0,pin1_light_T0_preamp,'go','MarkerFaceColor','g')
 loglog(irradiance1550_T0,pin2_light_T0_preamp,'bo','MarkerFaceColor','b')
 loglog(irradiance1550_T0,pin3_light_T0_preamp,'mo','MarkerFaceColor','m')
 title({'T = 0°C '}, 'fontsize', 16);
 xlabel('Irradiance (W/m^2)', 'fontweight', 'bold', 'fontsize', 16);
 ylabel('Vout (LIGHT) [V]','fontweight', 'bold', 'fontsize', 16);
 ylim([10E-5 10E-2]);
 lgd = legend('Pin0','Pin1','Pin2','Pin3');
 lgd = legend('Location','southeast');
 set(gca,'FontSize',16); grid on;

#Vout(offset corrected) vs. irradiance @temperature = 0°C
#Vout(offset corrected) = Vout(light) - mean(Voffset(dark))
pin0_out_T0_preamp = pin0_light_T0_preamp - mean(pin0_dark_preamp(T0));
pin1_out_T0_preamp = pin1_light_T0_preamp - mean(pin1_dark_preamp(T0));
pin2_out_T0_preamp = pin2_light_T0_preamp - mean(pin2_dark_preamp(T0));
pin3_out_T0_preamp = pin3_light_T0_preamp - mean(pin3_dark_preamp(T0));

figure(2); hold on;
 loglog(irradiance1550_T0,pin0_out_T0_preamp,'ro','MarkerFaceColor','r')
 loglog(irradiance1550_T0,pin1_out_T0_preamp,'go','MarkerFaceColor','g')
 loglog(irradiance1550_T0,pin2_out_T0_preamp,'bo','MarkerFaceColor','b')
 loglog(irradiance1550_T0,pin3_out_T0_preamp,'mo','MarkerFaceColor','m')
 title({'T = 0°C '}, 'fontsize', 16);
 xlabel('Irradiance (W/m^2)', 'fontweight', 'bold', 'fontsize', 16);
 ylabel('Vout (offset corrected) [V]','fontweight', 'bold', 'fontsize', 16);
 ylim([10E-6 10E-3]);
 lgd = legend('Pin0','Pin1','Pin2','Pin3');
 lgd = legend('Location','southeast');
 set(gca,'FontSize',16); grid on;

#Responsivity vs. irradiance @temperature = 0°C
optical_power_T0 = sample_width * sample_length * irradiance1550_T0;

responsivity0_T0 = pin0_out_T0_preamp./optical_power_T0;
responsivity1_T0 = pin1_out_T0_preamp./optical_power_T0;
responsivity2_T0 = pin2_out_T0_preamp./optical_power_T0;
responsivity3_T0 = pin3_out_T0_preamp./optical_power_T0;

figure(3); hold on;
 loglog(irradiance1550_T0,responsivity0_T0,'ro','MarkerFaceColor','r')
 loglog(irradiance1550_T0,responsivity1_T0,'go','MarkerFaceColor','g')
 loglog(irradiance1550_T0,responsivity2_T0,'bo','MarkerFaceColor','b')
 loglog(irradiance1550_T0,responsivity3_T0,'mo','MarkerFaceColor','m')
 title({'T = 0°C '}, 'fontsize', 16);
 xlabel('Irradiance (W/m^2)', 'fontweight', 'bold', 'fontsize', 16);
 ylabel('Responsivity (V/W)','fontweight', 'bold', 'fontsize', 16);
 ylim([10E7 10E10]);
 lgd = legend('Pin0','Pin1','Pin2','Pin3');
 lgd = legend('Location','southeast');
 set(gca,'FontSize',16); grid on;

#SEI vs. irradiance @temperature = 0°C
i0 = temp_dark == 0;
pin0_dark_T0_std = std(pin0_dark(i0));  pin0_dark_T0_std_preamp = pin0_dark_T0_std/20;
pin1_dark_T0_std = std(pin1_dark(i0));  pin1_dark_T0_std_preamp = pin1_dark_T0_std/20;
pin2_dark_T0_std = std(pin2_dark(i0));  pin2_dark_T0_std_preamp = pin2_dark_T0_std/20;
pin3_dark_T0_std = std(pin3_dark(i0));  pin3_dark_T0_std_preamp = pin3_dark_T0_std/20;

SEI0_T0 = pin0_dark_T0_std_preamp./(responsivity0_T0 * sample_width * sample_length);
SEI1_T0 = pin1_dark_T0_std_preamp./(responsivity1_T0 * sample_width * sample_length);
SEI2_T0 = pin2_dark_T0_std_preamp./(responsivity2_T0 * sample_width * sample_length);
SEI3_T0 = pin3_dark_T0_std_preamp./(responsivity3_T0 * sample_width * sample_length);
```