Predictive Multiple Sampling Algorithm with Overlapping Integration Intervals for Linear Wide Dynamic Range Integrating Image Sensors

by

Pablo M. Acosta-Serafini

Electric/Electronic Engineer
Universidad Católica de Córdoba (Argentina), November 1994

Master of Science in Electrical Engineering and Computer Science
Massachusetts Institute of Technology, June 1998

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Electrical Engineering and Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

February 2004

Author . . . .
Department of Electrical Engineering and Computer Science
January 30, 2004

Certified by . . . .
Charles G. Sodini, Ph.D.
Professor of Electrical Engineering
Thesis Supervisor

Accepted by . . . .
Arthur C. Smith, Ph.D.
Chairman, Committee on Graduate Students
Department of Electrical Engineering and Computer Science

Predictive Multiple Sampling Algorithm with Overlapping Integration Intervals for Linear Wide Dynamic Range Integrating Image Sensors

by

Pablo M. Acosta-Serafini

Submitted to the Department of Electrical Engineering and Computer Science
on January 30, 2004, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Electrical Engineering and Computer Science

Abstract

Machine vision systems are used in a wide range of applications such as security, automated quality control and intelligent transportation systems. Several of these systems need to extract information from natural scenes in the section of the electromagnetic spectrum visible to humans. These scenes can easily have intra–frame illumination ratios in excess of $10^6 : 1$. Solid–state image sensors that can correctly process wide illumination dynamic range scenes are therefore required to to ensure correct reliability and performance.

This thesis describes a new algorithm to linearly increase the illumination dynamic range range of integrating–type image sensors. A user–defined integration time is taken as a reference to create a potentially large set of integration intervals of different duration (the selected integration time being the longest) but with a common end. The light intensity received by each pixel in the sensing array is used to choose the optimal integration interval from the set, while a pixel saturation predictive decision is used to overlap the integration intervals within the given integration time such that only one frame using the optimal integration interval for each pixel is produced. The total integration time is never exceeded. Benefits from this approach are motion minimization, real–time operation, reduced memory requirements, programmable light intensity dynamic range increase and access to incremental light intensity information during the integration time. The algorithm is fully described with special attention to the resulting sensor transfer function, the signal–to–noise ratio, characterization of types and effects of errors in the predictive decision, calculation of the optimal integration intervals set given a certain set size, calculation of the optimal number of integration intervals, and impact of the new algorithm to image data compression.

An efficient mapping of this algorithm to a CMOS process was done by designing a proof–of–concept integrated circuit in a 0.18μm 1.8V 5–metal layer process. The major components of the chip are a 1/3" VGA (640×480) pixel array, a 4bit per pixel memory array, an integration controller array and an analog–to–digital converter/correlated double sampled (ADC/CDS) array. Supporting components include pixel and memory row decoders, memory and converter output digital multiplexers, pixel–to–ADC/CDS analog multiplexer and test structures. The pixels have a fill factor of nearly 50%, as most of the needed system additions and complexity were taken off–pixel. The prototype is fully functional and linearly expands the dynamic range by more than 60dB.

Thesis Supervisor: Charles G. Sodini, Ph.D.
Title: Professor of Electrical Engineering

Acknowledgments

I would like to thank my adviser, Prof. Charles Sodini for his excellent technical guidance throughout this project, for his support, encouragement and patience. He was always interested in me as a student, as an engineer, and as a person. For all of this and much more I have an enormous debt of gratitude.

The members of the Sodini Research Group are responsible in no small measure for my success. Don Hitko and Dan McMahill were always willing to help, teach, fix and explain anything that I asked them. The time they always graciously spent with me will always be remembered. My buddies, Ginger Wang and Iliana Fujimori, greatly contributed to my project in our weekly imaging meetings, in our constant technical discussions, and even more importantly, in our constant escapades to Toscanini's. I doubt that the heights of knowledge and ice–cream nirvana we reached will ever be duplicated. The new batch of recruits, specially John Fiorenza, Andy Wang, Lunal Khuon, Anh Phan, Todd Sepke and Farinaz Edalat contributed to a fun and relaxed working environment, and never complained when the lab was perpetually kept in the dark during my measurements.

The research was funded by a National Semiconductor fellowship, by the member companies of the MIT Center for Integrated Circuits and Systems, and of the MIT MTL Intelligent Transportation Research Center. The proof–of–concept integrated circuit was fabricated by National Semiconductor. Besides their CAD tools and computer resources, the assistance of the technical staff in the Salem, NH design center was invaluable.

Souren (Sam) Lefian was instrumental during the integrated circuit testing. His wide range of skills was constantly needed during this protracted phase. He was always eager to help, listen, do and commiserate with me at a moment's notice.

My family was my personal cheering section, the supporting rock upon which I rested, unquenchable fountain of solace, humor, deserved prodding, and always, source of inspiration. I hereby nominate for sainthood my parents Eduardo and Estela, my Godmother Aunt Chuly, my brother Eduardo and my sister–in–law Valeria. To my young, energetic and inquisitive nephews and Godchildren Alejandro and Marcos, the future is truly yours!

The end of this journey is yet another sign of God's hand in my life. His arms carried me during the dark moments, and he granted me more happiness and peace than I deserved in the past few years. Blessed be his holy name forever.

Dedication

To my family

A mi familia

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Solid–state image sensors have become part of everyday life. Ubiquitous in a wide range of consumer applications, they are also a critical component in various machine vision systems. Automated quality control [1,2], security [3,4] and emerging intelligent transportation systems [5–7] are only a handful of examples where visible light image sensors constitute the interface between the real world and processing elements.

A machine vision system typically includes an image sensor, which nowadays provides digital output, and a signal processor which also handles the data in the digital domain [8] (Figure 1-1). The natural scenes which these systems have to process can have light intensity ratios exceeding $10^6 : 1$ [9–11], so image sensors that meet this requirement are critical for adequate performance and reliability. Additionally, other applications such as scientific research (astronomical telescopes, biological cell filming, etc.) and high–end consumer cameras stand to benefit from advances and improvements in the intensity dynamic range of image sensors.

The implications of a bounded dynamic range, as well as the challenges involved in
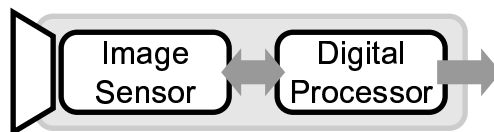
Figure 1-1: Typical machine vision system.

extending it, have long been identified. Consequently there have been numerous attempts to solve the problem, and several techniques have been developed. Since increasing spatial resolution is a constant driver for all areas of imaging (machine vision, scientific and consumer), techniques that minimize the impact on pixel design and transistor count are preferred. Also, since most of the current methods to implement color processing and other image processing tasks rely on a linear irradiance transfer characteristic, imagers that achieve wide dynamic range in linear fashion are preferred, since then there is no need to account for (and remove) the non–linearity of the sensor. This saves processing power, processing time and does not reduce the pixel output resolution. Additionally, linear sensors are also preferred because they preserve details even for high illumination regions.

The goal of this thesis is to demonstrate that the light intensity dynamic range of integrating–type image sensors can be linear and dramatically increased in an efficient manner, without major alterations to existing sensing arrays.

1.2 Thesis Contributions

The main contribution of this thesis is the development of a predictive multiple sampling algorithm to increase the upper bound of integrating–type image sensors. The predictive decision was used to arrange integration times of different duration so that they would have a common end. In this way the dynamic range expansion afforded by the multiple sampling technique is maximized because the longest integration time matches the total integration time. The algorithm was fully described and its performance fully characterized. In this area the novel results that are portable to other multiple sampling techniques include:

- A technique to find the optimal set of integration times given a certain (fixed) number of them. This technique is based on a new, computationally–friendly formula to evaluate the intensity–to–digital (I2D) code quantization error. This expression can also be used to evaluate whether adding integration times reduces the I2D quantization error for given image statistics.

- The relationship between the resolution of the analog–to–digital converter(s) (ADCs) used to quantize the pixel output and the set of integration times used, and how this affects the monotonicity of the sensor transfer characteristic.

- The effects on image compression resulting from the unique transfer characteristic of multiple sampling sensors.

The CMOS proof–of–concept imager showcases a key advantage of the novel multiple sampling algorithm, the creation of a framework for significant dynamic range increase that can be efficiently implemented. This is highlighted in 1) a pixel design that requires minimal alterations to the standard three–transistor cell, which results in good fill factors and allows the pixels to be suitable for sensing arrays of large spatial resolution, 2) shift of the complexity to the column or system level and 3) simple implementation of an automatic brightness adaptation mechanism. While the design of the supporting structures (analog–to–digital converters, SRAM cells, etc.) did not include new techniques, two blocks are new and relevant of mention:

- A compact integration controller for predictive multiple sampling algorithms. A custom 4–bit digital summer, 4–bit digital comparator, a "D"–type flip–flop and a dynamic comparator are the only major blocks needed. The design is small enough that it could be used to minimize pixel size when pixel–parallel performance is required.

- A M–to–N analog multiplexer. While the decision to have 64 ADCs instead of full column–parallel ADC array significantly increased the complexity of the system timing and affected system performance, it did create the need for a compact, efficient analog multiplexer able to route the 1920 pixel output lines to the 64 converters. While developed specifically for the proof–of–concept integrated circuit, this multiplexer can be used anywhere analog multiplexing is needed, being particularly suited for situations where a large number of input channels have to be routed to a relatively small number of output channels. As the number of input channels (columns in a imager) increase, the use of a standard pass gate chain decoding multiplexer becomes suboptimal. Extra parasitic resistive and capacitive elements are unnecessarily added to potentially sensitive analog nodes, and settling time is compromised. The new structure brings the number of pass gates needed to its bare minimum (one) and does so in an efficient, compact way, as only a "D"–type flip–flop is needed per input channel (M). This also allows the multiplexer to be used in designs where tight inter–channel spacing (like an imaging array) is required. Selecting channels is as easy as loading a shift–register. Finally, the layout of this multiplexer does not significantly scale with

the number of output channels (N), only added interconnect is required.

1.3   Thesis Organization

This chapter briefly stated the need for wide dynamic range image sensors in machine vision systems. Chapter 2 provides an overview of the different dynamic range increase methods previously reported. Due to their overwhelming popularity, most of the imagers presented use CMOS technology, though charge–coupled devices (CCD) are also presented when they show techniques unique to this technology. The basics of the multiple sampling technique for integrating–type image sensors can also be found in this chapter.

Chapter 3 delves into the specifics of the multiple sampling dynamic range increase method developed as part of this thesis. Its particular implementation of the general concept is reviewed, as well as its performance, features and benefits. A novel technique to obtain the optimal duration of integration times given a fixed number of them can be found in this chapter. The effects that the new algorithm has on image compression are also discussed.

Chapter 4 covers the design of a proof–of–concept image sensor. This integrated circuit has a VGA sensing array, 64 on–chip analog–to–digital converters, on–chip per–pixel digital memory and column–parallel integration control.

Chapter 5 presents the experimental results obtained from the proof–of–concept imager. Relevant measurements include the sensor transfer characteristic, signal–to–noise ratio and noise. Sample frames showing the new algorithm in action can be seen in this chapter.

Chapter 6 summarizes the results and contributions of this thesis. Directions for future work in the area of wide dynamic range image sensors are suggested.

Chapter 2

Background

2.1 Illumination Dynamic Range Overview

A typical image sensor is made out of pixels, arranged in a two–dimensional array, that capture light coming from a scene and convert it to an electrical quantity (charge, voltage or current). When incident photons have energy in excess of the bandgap of the semiconductor, they create electron–hole pairs that are separated by the electric field present in the pixel's photodiode. Only one type of carrier is preserved (typically electrons), the other is discarded to the substrate.

Imagers can be broadly classified in two types, continuous–time or integrating, based on when the illumination signal can be read. In the former, the illumination signal can be read almost instantaneously. In continuous–time pixels, the output is typically either the photocurrent itself (possibly amplified) or a voltage when a transconductance element is present. Power supply levels, maximum power dissipation, area and other technology, system and design factors limit the maximum photocurrent or photovoltage level.

Integrating image sensors only produce a valid output at predefined intervals. Pixels in this type of sensors allow a photodiode to accumulate (integrate) photo–generated charge for a period of time after a reset cycle (Figure 2-1) [12]. The pixel output, which can be in any electrical domain, is a scaled version of the photodiode charge. This output can later be quantized by an ADC to produce a digital sample that represents the average light intensity received by the photodiode during the integration time. The pixel signal[1] (S) is proportional to the light intensity (I) and the integration time (T_{INT}) as long as the pixel signal stays

---

[1]Pixel output change resulting from a charge change in the photodiode during the integration time.
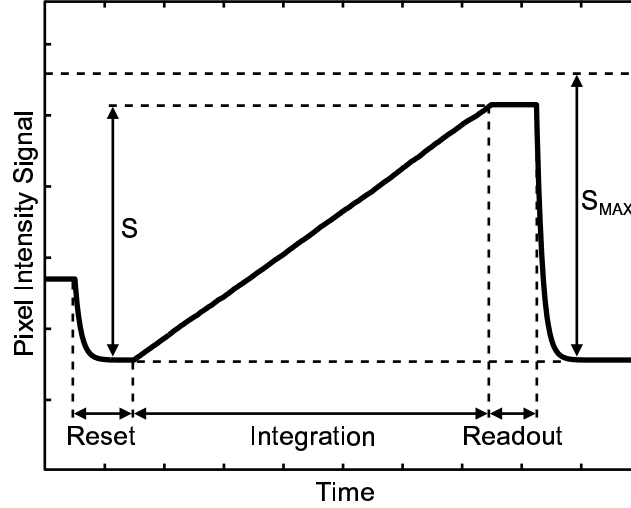
Figure 2-1: Typical pixel cycle in an integrating–type image sensor receiv-
ing a constant light intensity. An initial reset cycle zeroes the
intensity signal, which later increases linearly throughout the
integration time.

below its saturation point (S_{MAX}). This limit can be set by the photodiode itself or by
the pixel read–out circuitry. When the intensity received by the photodiode is such that
the pixel signal saturates during the integration time, further photo–generated charge does
not produce a proportional output, $S \approx S_{MAX}$ for the remainder of the integration time
and consequently there is loss of visual information. For a given saturation point a finite
integration time defines two intensity ranges:

$$\forall \; (T_{INT}, S_{MAX}) \, \exists \, I_{TH} : \left\{ \begin{array}{llll} I & \in & [0, I_{TH}) & , S < S_{MAX} \\ I & \in & [I_{TH}, \infty] & , S = S_{MAX} \end{array} \right. \tag{2.1}$$

There is no loss of visual information as long as the intensity received is below the thresh-
old I_{TH}, which is proportional to the saturation point and inversely proportional to the
integration time.

The illumination signal is thus limited by fabrication process parameters, system factors
and circuit implementation regardless of the type of pixel used. Additionally, the noise of the
pixel readout circuitry and other noise sources overwhelm the illumination signal in low light
situations. The ratio between the illumination that saturates the pixel and the minimum
detectable illumination is defined as the dynamic range of the image sensor (Figure 2-2).

Dynamic range expansion can therefore be achieved by reducing the noise floor, en-
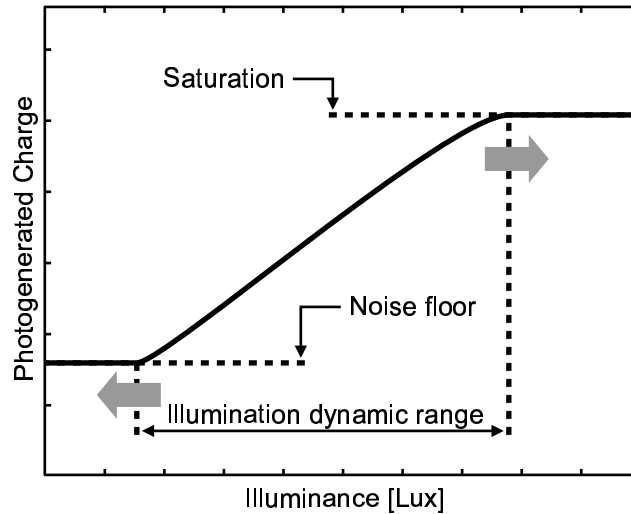
Figure 2-2: Dynamic range of a typical image sensor. Arrows indicate direction of improvement efforts
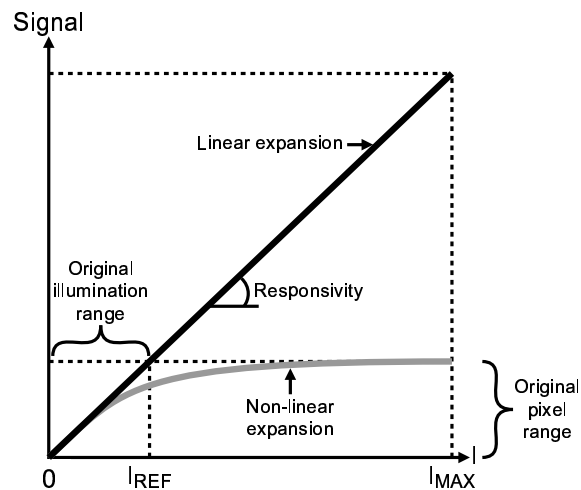
Figure 2-3: Linear and nonlinear image sensor transfer characteristics resulting from different wide dynamic range approaches.

abling the capture of darker scenes, and/or by increasing the saturation point, enabling the capture of brighter scenes. Low noise circuitry, careful engineering of the photodiode reverse saturation current and other improvements can help lower the noise floor, while the increase of the saturation level, the subject of this thesis, has been attempted using a host of different techniques. Some of these techniques only expand the illumination range without expanding the pixel signal range accordingly (Figure 2-3). The resulting nonlinear transfer characteristics have decreased responsivity (first derivative of the transfer characteristic)

which leads to decreased contrast and loss of details at high illuminations [13]. Additionally, the nonlinearity typically has to be canceled before color processing and other image processing tasks can be performed, since most of these are linear [14].

2.2   Reported Saturation Level Increase Techniques

The different approaches taken to increase the pixel saturation level have resulted in several types of imagers [13, 15]. Wide dynamic range techniques applied to continuous–time sensors can be seen in:

- Logarithmic sensors.

- Multimode sensors.

Techniques applied to integrating sensors can be seen in:

- Clipped sensors.

- Frequency–based sensors.

- Multiple sampling sensors.

2.2.1   Logarithmic Image Sensors

This type of sensors compress the transfer characteristic using a continuous (logarithmic) function in order to acquire an extended illumination range without exceeding the original photodiode signal swing. The basic pixel design is shown in Figure 2-4: an n^+–p substrate reverse–biased photodiode generates a current which is converted to a voltage by an MOS transistor operating in the subthreshold regime [16, 17] or in weak inversion [18]. As the photogenerated carriers are collected in the transistor's source, the voltage at this terminal decreases logarithmically as some carriers are able to overcome the potential barrier of the channel [19]. It can be shown that the relationship between output voltage V_{OUT} and photodiode current I_{PH} is [16]:

$$V_{OUT} = V_{Th} \cdot ln \left(1 + \frac{I_{PH}}{I_0} \right) \tag{2.2}$$

(a) Transistor in sub-threshold region with $V_{GS} = 0$.

(b) Transistor in subthreshold region with $V_{GS} < V_T$ (V_T denotes the transistor threshold voltage).
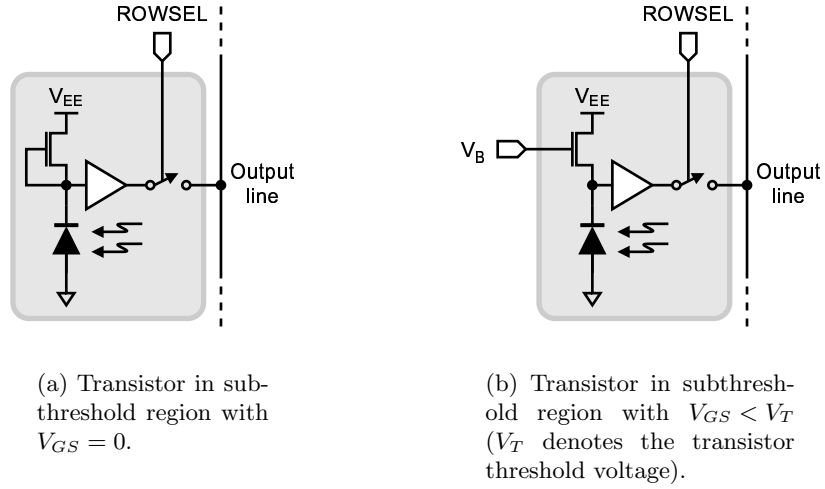
Figure 2-4: Logarithmic pixels.

where $V_{Th} = k \cdot T/q \approx 26\text{mV} @ T = 300°K$ is the thermal voltage and I_0 is the "off" current of the MOS transistor ($I_D @ V_{GS} = 0$). Then, for low light situations ($I_{PH} \ll I_0$):

$$V_{OUT} \approx V_{Th} \cdot \frac{I_{PH}}{I_0} \tag{2.3}$$

since $ln(1 + x) \approx x$ for $x \to 0$. This logarithm dependence has several implications:

a) It reduces the contrast and details in high illumination regions [13].

b) It limits the voltage swing at the photodiode node. For example, a 10^6 to 1 illumination change only generates a 350mV difference.

c) The nonlinear dependence on semiconductor parameters creates a nonlinear pixel–to–pixel fixed pattern noise (FPN)[2].

d) It makes the contrast ratio between two regions independent of their illumination level, matching how the human eye perceives natural scenes [14, 20].

Assuming that the background (large signal) illumination received by the pixel I_{BG} dominates over the leakage current I_L, the bandwidth of these basic logarithmic pixels is given by:

$$\tau_{PH} = \frac{C_{PH}}{g_m} \Rightarrow f_{3dB} = \frac{1}{2 \cdot \pi \cdot \tau_{PH}} = \frac{I_{BG}}{2 \cdot \pi \cdot V_{Th} \cdot C_{PH}} \tag{2.4}$$

---

[2]Fixed (constant) pixel–to–pixel output variation observed under spatially uniform illumination.

(a)   Phototransistor mode.
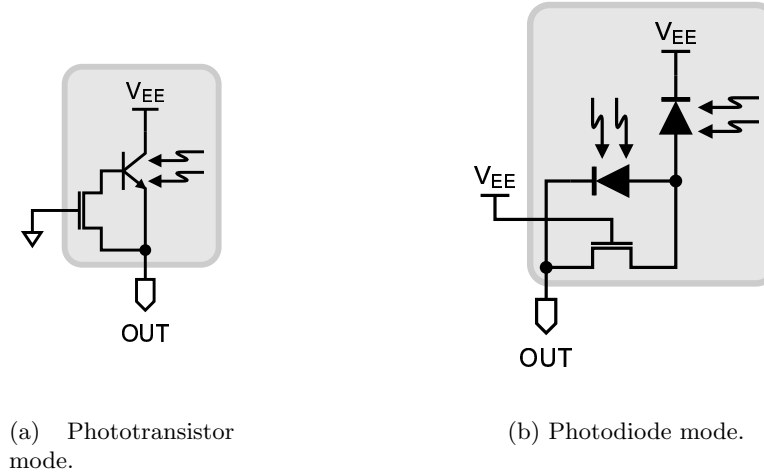
(b) Photodiode mode.

Figure 2-5: Multimode pixel.

where C_{PH} is the capacitance of the photodiode node and $g_m = I_{BG}/V_{Th}$ is the subthreshold transconductance of the MOS transistor [19]. As it can be seen, the frequency response is illumination–dependent, scenes with significant background illumination can be captured at high frame rates, while dark scenes can exhibit significant image lag.

The receptor shot noise of logarithmic sensors is independent of the illumination (signal) received:

$$\overline{v_{PH}^2} = \frac{\overline{i_{PH}^2}}{g_m^2} = \frac{2 \cdot q \cdot I_{BG} \cdot \Delta f}{g_m^2} = \frac{2 \cdot q \cdot I_{BG}}{g_m^2} \cdot \frac{\pi}{2} \cdot f_{3dB} \Rightarrow \overline{v_{PH}^2} = \frac{k \cdot T}{2 \cdot C_{PH}} \qquad (2.5)$$

where $\Delta f = (\pi/2) \cdot f_{3dB}$ is the equivalent noise bandwidth of the system [21]. Variations and refinements of this basic pixel design have been extensively reported [22–26].

2.2.2   Multimode Image Sensors

This approach is inspired by the human eye, where two types of cells with different sensitivity (rods and cons) are used to capture a wide range of illuminations using an adaptive mechanism [14]. Multimode sensors combine the most commonly used photodetectors in a single pixel: a vertical bipolar junction phototransistor (BJT) with high sensitivity but larger FPN[3], and a photodiode, which is comparatively relatively insensitive to light [27, 28].

---

[3]The current gain β of the phototransistor in general is different from pixel to pixel due to variations in the fabrication process parameters.
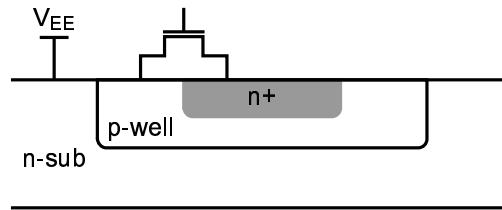
Figure 2-6: Multimode pixel cross–section.

Figure 2-5 shows the two modes of operation schematically. In the reported sensor an n–substrate process is used, so the lateral BJT is formed by a substrate–p–well–n$^+$ diffusion structure (Figure 2-6). The photodiode is made by the p–well and n$^+$ diffusion. An NMOS transistor whose drain and source are connected to the p–well and n$^+$ diffusion is the device that actually selects the operating modes: if the transistor is on the photodiode mode is selected as the base and emitter of the BJT are shorted out, while if the transistor is off the BJT mode is selected.

The basic multimode structure can be cascaded, in Darlington fashion, to provide extra amplification and sensitivity for low illuminations. If this is done, the resulting pixel has three modes of operation –Darlington, BJT and photodiode– encoded in a two–bit digital bus controlling the NMOS switches. The actual decision of which mode to use has to be taken by an adaptive mechanism, whose goal is to keep the resulting pixel photocurrent within the range of downstream signal processing circuits [27]. Real–time local brightness adaptation was achieved by a 16–transistor controller which commands a 4–pixel cell [29].

2.2.3 Clipped Image Sensors

In this type of sensors the rate of photogenerated carrier accumulation is controlled either continually or at predefined intervals. One of the first clipped pixel designs can be seen in Figure 2-7. In this photogate implementation a traditional anti–blooming gate, which is here kept at a slight potential offset with respect to the imaging gate, is used to provide dynamic range expansion [30, 31]. The photogenerated carriers are initially accumulated under the imaging gate, but at higher illumination levels some of the carriers are injected over the anti–blooming gate potential barrier (not unlike logarithmic pixels) thus slowing the accumulation process. The charges are subsequently transferred to a floating diode readout circuitry at the end of the integration time. The number of carriers accumulated
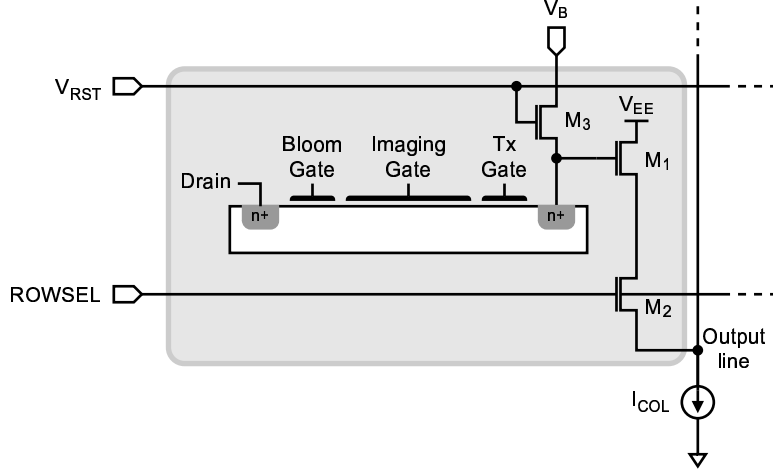
Figure 2-7: Photogate implementation of a clipped pixel.

in the imaging gate N_{PH} is [30]:

$$N_{PH} = \begin{cases} \frac{I_{PH} \cdot T_{INT}}{q} & \text{, Log acc. negligible.} \\ \frac{C_{PH} \cdot V_{IB}}{q} + \frac{C_{PH} \cdot V_{Th}}{q} \cdot ln\left(\frac{I_L + I_{PH}}{I_L + I_{PH} \cdot e^{-\frac{(I_L + I_{PH}) \cdot T_{INT}}{C_{PH} \cdot V_{Th}}}}\right) & \text{, Log acc. significant.} \end{cases}$$

$$(2.6)$$

where V_{IB} is the potential difference between the imaging and anti–blooming gates. Since
the time constant of the logarithmic operation is inversely proportional to the photocurrent
(Equation 2.4), for short integration times ($T_{INT} \ll \tau_{PH}$) the photocurrent has to be very
large for the logarithmic compression to be significant. Therefore, in these circumstances the
charge accumulation is mainly linear. However, for longer integration times ($T_{INT} \gg \tau_{PH}$),
the logarithmic accumulation mode dominates and the pixel transfer characteristic is com-
pressed.

The pixel transfer function also has a strong dependence on the potential barrier V_{IB}
between the imaging and anti–blooming gates [30]:

$$V_{IB} = V_{Th} \cdot ln\left(\frac{I_0}{I_L}\right) + V_{I_0} - V_B \tag{2.7}$$

where V_{I_0} is the potential of the imaging gate for no illumination and V_B is the anti–blooming
gate potential. As it can be seen, the potential barrier has the same dependence as the
time constant of the logarithmic accumulation period, therefore for a fixed integration time
T_{INT} both time constant and barrier are small for a large leakage current I_L, making the
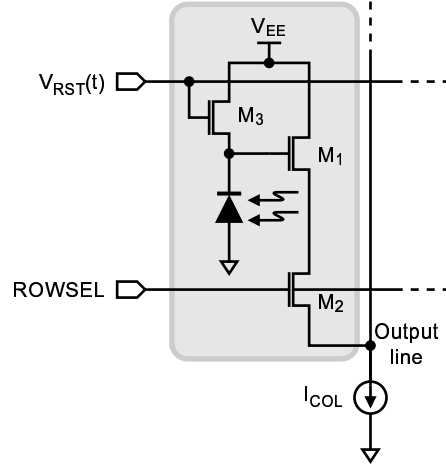
Figure 2-8: CMOS implementation of a clipped pixel based on the barrier stepping concept.

logarithmic accumulation dominant. On the other hand a small leakage current makes the linear accumulation dominant for a fixed integration time.

Since this pixel can be completely reset, image lag is reduced and reset noise is eliminated. Also, since the subthreshold operation of the anti–blooming gate does not depend on its drain potential, higher uniformity across the sensing array can be achieved.

A CMOS implementation of the clipping concept builds upon a standard active pixel cell (Figure 2-8) [32]. In this case V_{RST}, the potential at the gate of the reset transistor $M3$, changes during the integration time. The reset cycle of the pixel sees V_{RST} at its highest potential, allowing excess charge to flow into the drain of $M3$. Then, during the integration, V_{RST} is systematically lowered so that the potential barrier between the integrating node and the drain of $M3$ is raised several times. Therefore, for some illuminations there are periods of time when the accumulated charge can increase linearly, but then there are also periods of time when the charge is limited by the reset transistor (Figure 2-9). With this technique, any compressive transfer characteristic can be approximated by a piecemeal linear function resulting from the timing of $V_{RST}(t)$.

The noise at the sensing node in this case has contributions from the clipped accumulation period and from the linear ("free") accumulation period [32]:

$$\overline{v_{PH}^2} = 2 \cdot \frac{k \cdot T}{C_{PH}} + \frac{q \cdot N_{Free}}{C_{PH}^2} \tag{2.8}$$
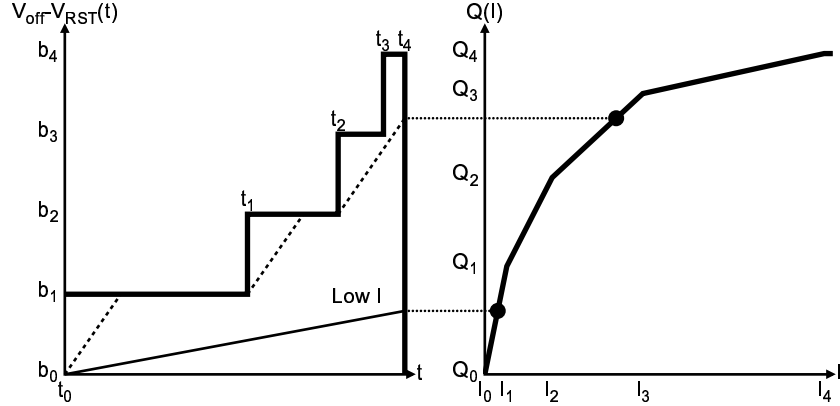
Figure 2-9: Sample compressed transfer characteristic of the CMOS clipped pixel
implementation based on the barrier stepping concept.

where N_{Free} denotes the number of photogenerated carriers accumulated during the lin-
ear period. The noise during the clipped accumulation region is essentially the same as
Equation 2.5[4], while the shot noise density during free/linear accumulation is simply pro-
portional to the number of photogenerated carriers.

Another CMOS implementation uses an in–pixel capacitor to limit the charge in the
photodiode (Figure 2-10 (a)) [33]. After a first exposure time T_1 the control line `TRANSFER` is
pulsed, prompting a charge redistribution between the photodiode and the storage capacitor.
A second, much shorter, exposure time T_2 follows, after which the photodiode voltage is
read. With equal photodiode and storage capacitances C_{PH}, the signal at the end of the
integration time is:

$$V_{OUT} = \begin{cases} \frac{1}{C_{PH}} \cdot \left(\frac{Q_{MAX}}{2} + Q_2\right) & \text{, Charge overflows.} \\ \frac{1}{C_{PH}} \cdot \left(\frac{Q_1}{2} + Q_2\right) & \text{, Charge does not overflow.} \end{cases} \tag{2.9}$$

where Q_{MAX} denotes the maximum charge that can be accumulated in the photodiode, Q_1
denotes the charge accumulated during the first exposure time and Q_2 denotes the charge
accumulated during the second exposure time. Since $T_1 > T_2$ the resulting transfer char-
acteristic is made of two linear sections, a higher responsivity section for low illuminations
and a lower responsivity section for bright illuminations (Figure 2-10 (b)).

A   current–mode   approach   of   the   clipping   concept   can   be   seen   in

---

[4]The factor of 2 difference is due to the correlated double sampling (CDS) operation needed to remove
the reset level from the pixel output.

(a) Pixel schematic.

(b) Sample transfer characteristic.
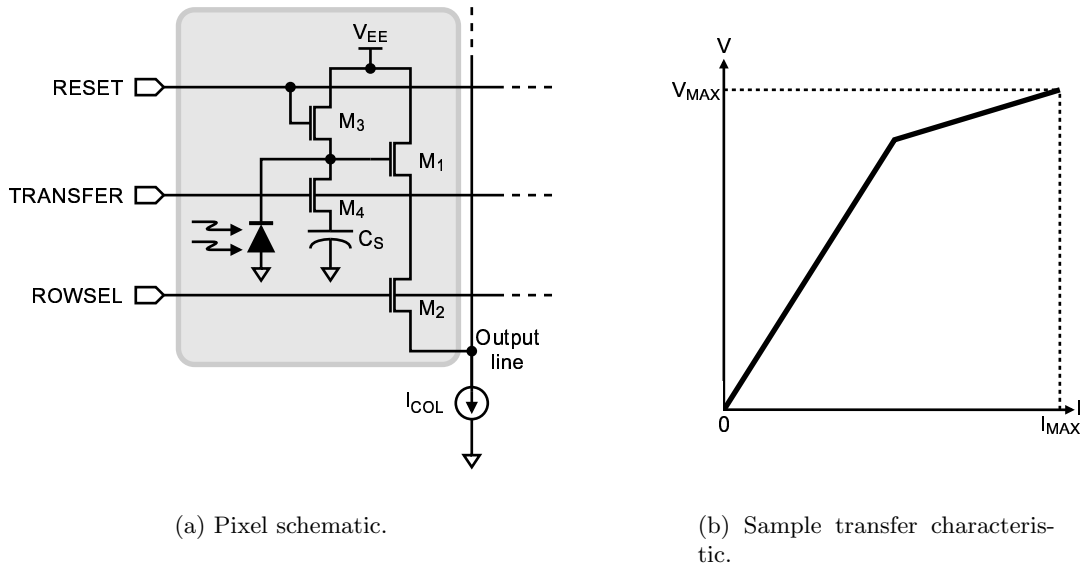
Figure 2-10: CMOS clipped pixel based on the in–pixel charge redistribution concept.

(a) Conceptual implementation.

(b) Suggested CMOS implementation with on–pixel offset storage.
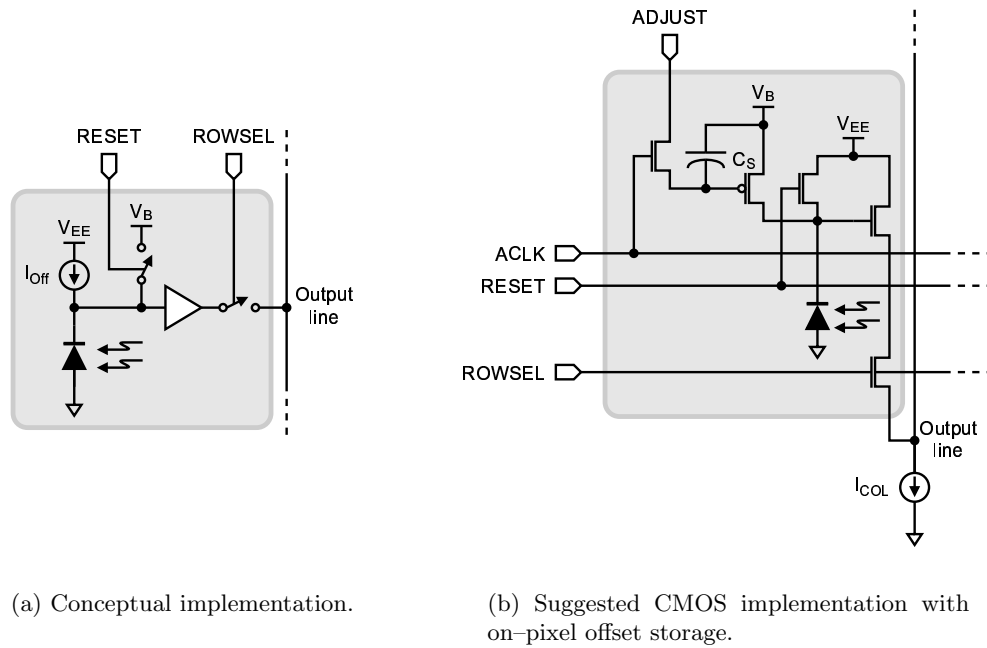
Figure 2-11: Current–mode clipped pixel.

Figure 2-11 (a) [34]. A scene may have a large background illumination I_{BG} which takes up most of the available integrated charge, leaving the relevant small signal illumination component hidden in the noise floor. By subtracting an offset current I_{Off} closely matched
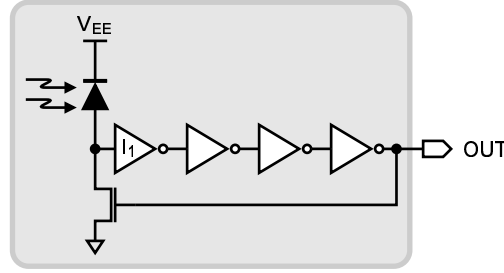
Figure 2-12: Frequency–based pixel using a digital inverter chain.

to the photocurrent generated by I_{BG}, most of the voltage difference in the pixel can now be the result of the small signal illumination. Several decades of illumination can then be correctly imaged by judiciously changing the offset current.

A suggested CMOS implementation can be seen Figure 2-11 (b) [34]. Since the photocurrents typically have a small magnitude, a PMOS transistor in weak inversion was used to implement the offset current source. Additionally, as in this regime the drain current is exponentially related to the gate–source potential, several decades of current can be obtained with only a small voltage change. The bias for this current source is stored in a capacitor present in the pixel, therefore the offset can be different from pixel to pixel, a feature that enables local brightness adaptation.

This pixel design may suffer from a small fill factor due to the presence of the storage capacitor and also because both types of devices (NMOS and PMOS) are used. Offset current mismatches are also possible due to kTC noise in the storage capacitor, which is significant due to the high sensitivity of the current to the gate–source potential. Real–time adaptation is not possible because prior knowledge of the illumination is needed to calculate the offset currents.

2.2.4 Frequency–based Image Sensors

In these type of sensors the illumination signal is transformed into a waveform with proportional frequency which can be detected and quantified over several decades. One scheme to achieve this is shown in Figure 2-12 [35]. The photodiode in this pixel is initially reset and then allowed to capture incident illumination. The voltage at the photodiode anode increases at a rate dependent on the illumination received and at some point in time it crosses the inverter $I1$ trip point. This event generates another reset cycle, and the whole
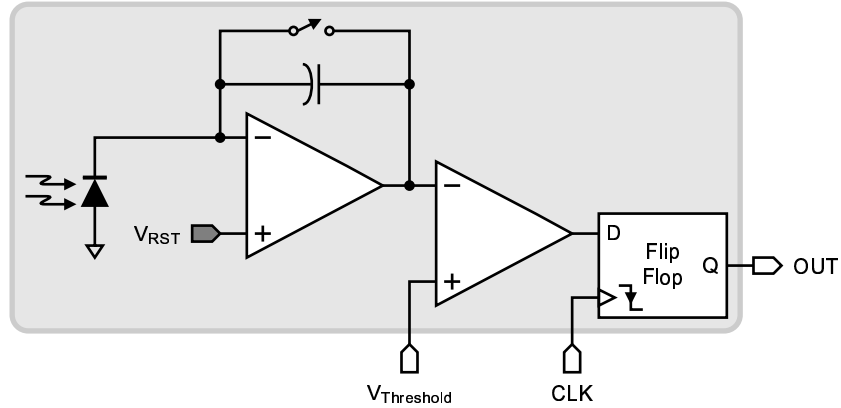
Figure 2-13: Frequency–based pixel using a single–slope ADC.

process is continually repeated, thus producing a square waveform at the pixel output whose frequency is directly proportional to the illumination. A chain of inverters is included in the pixel to allow sufficient time for the photodiode reset.

Pixel–to–pixel variations in the inverter trip points will introduce a non–linear fixed–pattern noise component. The pixel fill factor is reduced due to the presence of several NMOS and PMOS transistors. Since low illuminations translate into low frequencies, a long integration time may be needed to achieve adequate signal level resolution.

A refinement of this basic oscillating pixel uses an on–pixel Σ–Δ ADC to improve low light performance and decrease power dissipation [36–38]. Pixel fill factor is severely reduced due to large number of transistors needed to implement the quantizer.

Yet another frequency–based sensor uses a single–slope ADC to pin the photodiode voltage thus eliminating nonlinearities due to the voltage dependence of the photodiode capacitance (Figure 2-13) [39]. This pixel has both good low and high illumination perfor-mance, but it is only feasible for low spatial resolution sensors due to the large number of elements needed to implement the converter.

2.2.5 Multiple Sampling Image Sensors

While cost and fabrication technology cap the maximum pixel signal S_{MAX} of integrat-ing pixels, the integration time T_{INT} can be freely altered to modify the intensity threshold I_{TH}. Image sensors with illumination–dependent, but global, integration time address the pixel saturation issue but do not extend the intra–frame dynamic range [40–43]. The goal of the multiple sampling algorithm is to find the optimal integration time for every pixel

in the sensing array as a function of the intensity they receive. The optimal integration time is defined as the longest integration time for which the pixel does not saturate (refer to Section 3.5.5). Such an integration time can be found without prior knowledge of the intensity using the following procedure:

1. Integrate photo–generated charge using several (M) integration times of different duration (subsequently called integration slots) and record the pixel signal at the end of each slot[5]. The integration slot set ($\mathcal{T}$) and the pixel signal set ($\mathcal{S}$) are thus generated and can be expressed as:

$$\mathcal{T} \;=\; \{T_0, T_1, \ldots, T_{M-1}\}, \;\; T_0 > T_1 > \ldots > T_{M-1} \tag{2.10}$$

$$\mathcal{S} \;=\; \{S(T_0), S(T_1), \ldots, S(T_{M-1})\} \tag{2.11}$$

This step implicitly assumes that the intensity received by the pixel remains constant for all integration slots. If this is not the case, some or all of the elements in the pixel signal set are uncorrelated with each other and the optimality of the integration slot selection is not guaranteed.

2. Select the optimal integration slot i:

   - $i = 0$ if $S(T_0) < S_{MAX}$.

   - $i = M - 1$ if $S(T_{M-1}) = S_{MAX}$.

   - Otherwise:
   $$\exists \, i \in \{1, \ldots, M-1\} : \begin{cases} S(T_i) < S_{MAX} \\ S(T_{i-1}) = S_{MAX} \end{cases}$$

   As a consequence of the finite number of elements in the integration slot set, there exists an intensity range for which even the shortest integration slot cannot produce a non–saturated pixel signal. In this case the optimal integration slot is taken as the shortest one in set $\mathcal{T}$.

3. Calculate exposure ratio:

$$E_i = \frac{T_0}{T_i} \;\; \text{with} \;\; E_{MAX} = \frac{T_0}{T_{M-1}} \tag{2.12}$$

---

[5]$S(t)$ denotes the pixel signal at an elapsed time t after the end of the last reset cycle.
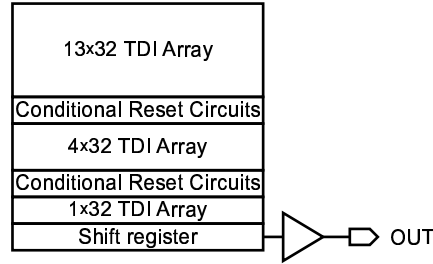
Figure 2-14: TDI CCD multiple sampling image sensor.

E_{MAX} is the increase in intensity dynamic range obtained by implementing the multiple sampling technique. The exposure ratio used by a given pixel will in general change from frame to frame depending on the intensity received. An associated exposure ratio set $(\mathcal{E})$ can be defined as:

$$\mathcal{E} = \{E_0, E_1, \ldots, E_{M-1}\} = \left\{1, \frac{T_0}{T_1}, \ldots, \frac{T_0}{T_{M-1}}\right\} \tag{2.13}$$

So for example, if every integration slot is half as long as the previous one, the exposure ratio set would be $\mathcal{E} = \{1, 2, 4, 8, \ldots\}$.

4. Produce the total pixel signal:

$$S_{TOT}(T_i) = E_i \cdot S_q(T_i) = \frac{T_0}{T_i} \cdot q(S(T_i)) \tag{2.14}$$

where $q(\cdot)$ denotes the quantizer function implemented by an analog–to–digital converter. The actual output format can be one of several options: the full resolution number $S_{TOT}(T_i)$, its two terms E_i and $S_q(T_i)$, the index i to the set $\mathcal{E}$ and $S_q(T_i)$, etc.

Different trade–offs can be made to implement the multiple sampling procedure, but from a machine vision standpoint the most important factor is to maximize E_{MAX} as given by Equation 2.12. One of the first reported attempts to implement this concept uses a time delay and integrate (TDI) CCD sensor where each pixel has 18 stages with conditional reset circuitry after 13, 4 and 1 stages (Figure 2-14) [44]. Therefore a pixel can integrate over 1, 5 or 18 stages depending on the illumination received. Each conditional reset stage compares the charge accumulated up to that point against a reference and discharges the
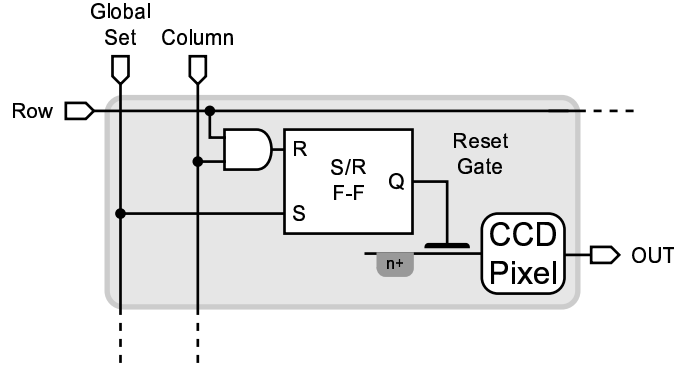
Figure 2-15: Multiple sampling CCD pixel with local brightness adaptation feature.

pixel if needed. The link between the pixel integration time and number of stages fixes the exposure ratio set $\mathcal{E}$ at fabrication time. Vertical spatial resolution is also limited by the large number of stages needed to achieve wide illumination dynamic range.

Another CCD implementation uses a reset gate (similarly placed as an anti–blooming gate) controlled by an in–pixel set/reset flip–flop (Figure 2-15) [45]. This register is initially set, inhibiting any charge accumulation under the imaging gate, and it is later reset to allow photogenerated charges to be collected for a period of time that depends on the illumination received in previous frames (frame–to–frame automatic adaptation is therefore not possible). A control chip is required to calculate the timing of the rest pulses for every pixel. Fill factor is reduced due to the need for digital circuitry in the pixel.

Other implementations strictly follow the basic algorithm and integrate photogenerated charges sequentially, one frame after another, while changing the integration time [46–48]. The optimal exposure for every pixel is then determined using all collected samples. While standard sensors with high fill factor pixels can be used with this methodology, massive systems storage is needed to save all the full–resolution samples for all the pixels in the sensing array.

For a given total integration time T_{INT} and shortest integration slot T_{M-1}, the maximum exposure ratio E_{MAX} (Equation 2.12) is maximized when the longest integration slot T_0 matches T_{INT}. An implementation that achieves this objective checks the pixels at predefined intervals $T_{INT}, T_{INT}/2, T_{INT}/4, T_{INT}/8, \ldots$ (Figure 2-16) [49]. During the first check the pixels are quantized and the first m most significant bits are obtained. Successive checks quantize the pixel again and can produce another m bits, but since the checking in-
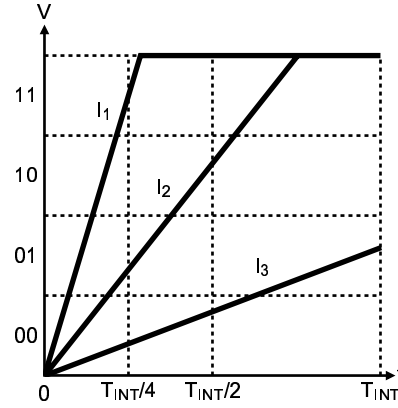
Figure 2-16: Multiple sampling image sensor producing Gray–coded output.

Table 2.1

Resulting digital code for sample illuminations

Illumination	b_3	b_2	b_1	b_0
I_1	1	1	1	1
I_2	0	1	0	1
I_3	0	0	0	1

terval is doubled, the signal doubles in magnitude too and thus only 1 bit of extra resolution is added. Therefore if the sensor allows for k checks, the pixel signal is quantized to $m + k$ Gray–coded bits (Table 2.1). This sensor needs to store (or transmit) the full–resolution output of every pixel as it is being produced. Blooming is possible since the pixels can remain saturated for almost all of the integration time.

Refinements to this checking scheme use local shuttering to stop the accumulation of photogenerated charges when it is predicted that the pixel is going to saturate (the basis for this decision is similar to the one detailed in Chapter 3) [50,51]. A schematic representation of this type of pixels can be seen in Figure 2-17. Leakage in the isolated node and diffusing photogenerated carriers from the photodiode can alter the stored value and consequently lead to errors when the final pixel output is produced.

This thesis describes a variant of the predictive multiple sampling method which improves, expands and systematizes efforts previously reported [52–54].
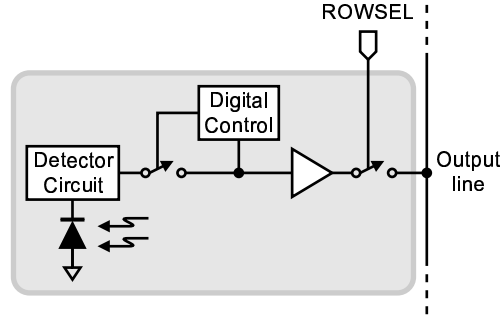
Figure 2-17: Multiple sampling pixel with local shuttering.

2.3   Summary

The illumination dynamic range of solid–state image sensors is a key factor in the reliability and performance of machine vision systems. Image sensors that can capture illumination ratios greater than 10^6:1 have been developed using a variety of techniques: logarithmic compression of the transfer characteristic, multimode operation, clipping of the accumulated charge, illumination–to–frequency conversion and multiple sampling.

Multiple sampling image sensors have emerged as a high performance option that can linearly increase the original pixel dynamic range. While this technique requires per–pixel storage which can add significant area to an existing imager, it provides a linear transfer characteristic which allows for normal image processing and has constant responsivity, preserving detail and contrast at high illuminations. For a given integration time the predictive variant of the multiple sampling algorithm, the focus of this thesis, maximizes E_{MAX}, the maximum dynamic range expansion ratio.

Chapter 3

Novel Algorithm for Intensity Range Expansion

3.1 Overview

The predictive multiple sampling algorithm greatly extends the light intensity dynamic range of existing image sensors. A user–defined integration time is taken as a reference to create a potentially large set of integration intervals of different duration (the selected integration time being the longest) but with a common end. The light intensity received by each pixel in the sensing array is used to choose the optimal integration interval from the set, while a pixel saturation predictive decision is used to overlap the integration intervals within the given integration time such that only one frame using the optimal integration interval for each pixel is produced. The total integration time is never exceeded. Benefits from this approach are:

- Motion minimization: artifacts due to object movement are not added since the integration time is not increased.

- Real–time operation.

- Reduced memory requirements: no intermediate frame(s) storage necessary.

- Programmable light intensity dynamic range increase.

- Access to incremental light intensity information during integration time.

- Minimal signal corruption: regardless of the particular integration interval used by a pixel, its signal is never held at an isolated node and can be quantized immediately after the end of the integration time.

A hardware implementation of the multiple sampling algorithm requires the addition of an integration controller and per-pixel memory to existing imager systems. Sensing arrays with non–destructive pixel read capabilities need to add a conditional pixel reset feature. Per–pixel storage is a fundamental requirement of the multiple sampling algorithm, as the determination of the optimal integration interval for a particular pixel receiving a given illumination occurs during the integration time, and this information is needed to calculate the total pixel output which only occurs after the integration time has ended. For performance reasons it is advisable to place the memory in the same silicon die as the sensing array, therefore the size of this on–chip memory (which typically is significant), constitutes the main cost of the dynamic range expansion.

3.2   Description

If the intensity is assumed to remain constant during the entire integration time, the pixel intensity signal increases linearly throughout it. Therefore pixel saturation can be predicted at any point during the integration time provided the pixel signal can be read without altering the photo–generated charge (non–destructive read). A destructive read would inject non–linearities in the accumulated charge which would eventually appear at the sensor output.

For a given integration slot of duration $T_j \in \mathcal{T}$, the intensity threshold I_{TH} (Equation 2.1) produces a linear pixel signal change[1]:

$$S_{TH}^{T_j}(t) = \frac{S_{MAX}}{T_j} \cdot (t - t_j) \quad , t \geq t_j \in [0, T_{INT}] \tag{3.1}$$

where t_j denotes the start of the integration slot used. This expression gives the signal threshold needed to predict pixel saturation. For example, Figure 3-1 shows a sensor with $\mathcal{T} = \{T_0\} = \{T_{INT}\}$ and two pixels that receive a different illumination intensity. If at any given time $t_a \in [0, T_{INT}]$ the pixel signal $S(t_a)$ is below $S_{TH}^{T_0}(t_a)$ the pixel will not saturate

---

[1]A linear charge–voltage relationship in the photodiode capacitance is assumed.
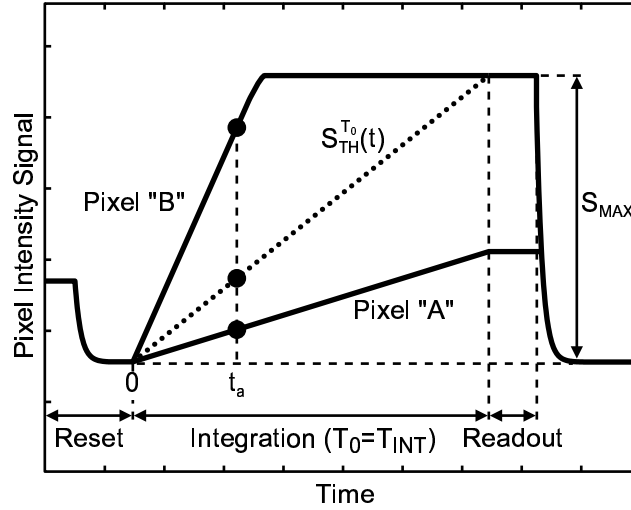
Figure 3-1: Intensity threshold used in the pixel saturation predictive decision. A pixel saturates at or before the end of the integration time if its signal is above the threshold $S_{TH}^{T_0}(t_a)$ at any point t_a during the integration time.

at the end of the integration time (Pixel "A"). If $S(t_a)$ is above $S_{TH}^{T_0}(t_a)$ then the pixel will saturate sometime before the end of the integration time (Pixel "B"). In the novel predictive algorithm the integration slots are temporally arranged to have a common ending with the longest integration slot matching the total integration time. At the (potential) beginning of each integration slot, a pixel check occurs. If saturation is predicted the pixel is reset and allowed to integrate for a shorter period of time (the next integration slot). If saturation is not predicted the pixel is allowed to integrate for the remainder of the current integration slot. In more precise terms (flow graph shown in Figure 3-2):

1. Select first integration slot by making $j = 0$.

2. If $j = M - 1$ integrate photo–generated charge for T_{M-1} and go to Step 6 (a regular integrating image sensor would therefore have $M = 1$). Otherwise integrate photo–generated charge for $T_j - T_{j+1}$ so as to be at the start of the $j+1$ integration slot, t_{j+1}.

3. Perform pixel check: if $S(t_{j+1}) \geq S_{TH}^{T_j}(t_{j+1})$ reset pixel (it is going to be saturated at the end of the j integration slot). Otherwise continue integrating photo–generated charge for T_{j+1} and go to Step 6.
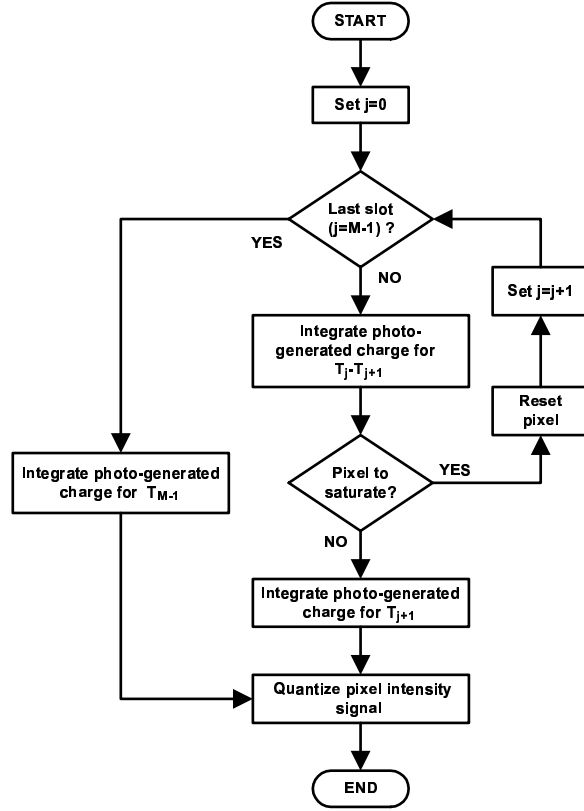
Figure 3-2: Novel predictive multiple sampling algorithm flow graph.

4. Select next integration slot by making $j = j + 1$.

5. Repeat procedure from step 2.

6. Quantize pixel intensity signal.

With this algorithm the optimal integration slot i is selected iteratively: on each check the algorithm decides whether the current integration slot j is the optimal one or if i is in the set of remaining slots $\{j + 1, \ldots, M - 1\}$. As a result, only the pixel intensity signal for the optimal integration slot $S(T_i)$ is generated.

3.3   Example

Figure 3-3 shows the behavior of two pixels in an image sensor that has $M = 2$ and $\mathcal{T} = \{T_0, 2/3 \cdot T_0, T_0/3\}$ with $T_0 = T_{INT}$. All the integration slots have a common ending and thus overlap toward the end of the integration time.
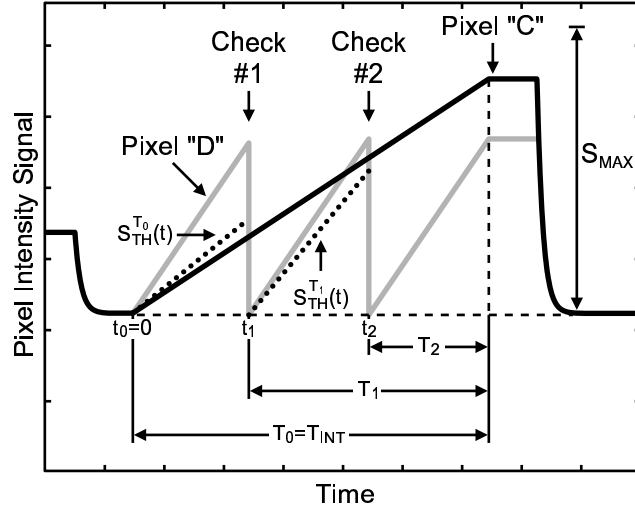
Figure 3-3: Novel predictive multiple sampling algorithm in action. Pixel "C" uses integration slot 0 and is never reset while Pixel "D" uses integration slot 2 and is reset twice.

Pixel "C" receives an intensity such that S_{MAX} is not reached even when the longest integration slot T_0 is used. Consequently when the first check cycle arrives its signal is below the threshold $S_{TH}^{T_0}(t_1)$ and the pixel is allowed to accumulate photo–generated charge for the remainder of T_0. Optimality of the integration slot selection requires no further checks because in some of them it might appear that the pixel is going to saturate (in a potential second check, $S(t_2) > S_{TH}^{T_1}(t_2)$). The total signal for Pixel "C" is $q(S(T_{INT}))$ as $E_i = E_0 = 1$.

Pixel "D" on the other hand receives a higher intensity and its signal is above the threshold at each of the two checks ($S(t_1) > S_{TH}^{T_0}(t_1)$ and $S(t_2) > S_{TH}^{T_1}(t_2)$) so the pixel is reset twice and the optimal slot, slot 2, is used to produce a non–saturated signal. The total signal for Pixel "D" is $3 \cdot q(S(T_2))$ as $E_i = E_2 = 3$.

3.4   Image Sensor Requirements

3.4.1   ADC resolution/integration slot ratios

The ADC resolution (N bits) and the integration slot set ($\mathcal{T}$) determine the monotonicity of the sensor transfer characteristic[2]. Since the transfer characteristic can be viewed as

---

[2]The analog–to–digital converter is assumed to be monotonic producing digital codes from 0 to $2^N - 1$.
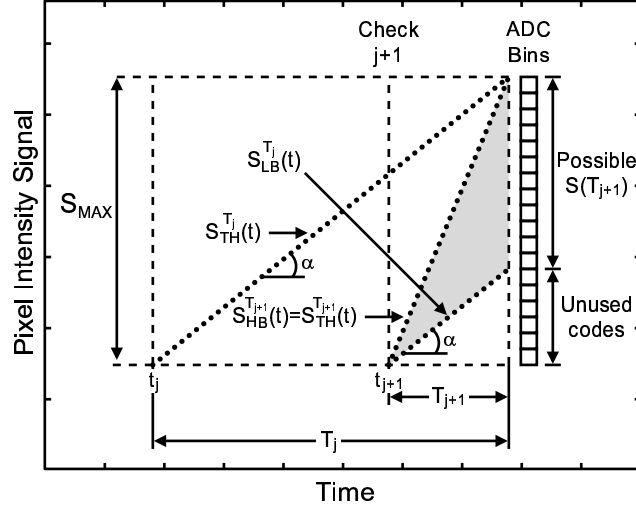
Figure 3-4: Relationship between the analog–to–digital converter resolution and the integration slots. Shaded area denote all possible $S(t)$ when the intensity received is such that integration slot $j+1$ is used.

made of different sections generated by the use of different integration slots, it is imperative that the starting digital code of one section is at least equal to the ending code of the preceding section. Figure 3-4 illustrates the situation where the pixel signal range of slot $j+1$ is bounded low by

$$S_{LB}^{T_{j+1}}(t) = S_{TH}^{T_j}(t - t_{j+1} + t_j) = \frac{S_{MAX}}{T_j} \cdot (t - t_{j+1}) \tag{3.2}$$

and bounded high by $S_{HB}^{T_{j+1}} = S_{TH}^{T_{j+1}}(t)$. Any intensity lower than $I_{TH}(T_j)$ uses an integration slot in the set $\{0, \ldots, j\}$ and any intensity higher than $I_{TH}(T_{j+1})$ uses an integration slot in the set $\{j+2, \ldots, M-1\}$. Since $S_{LB}^{T_j}(t)$ has a positive slope the pixel signal at the end of the integration slot $j+1$ is strictly positive $(S(T_{j+1}) > 0)$ and not all the pixel signal range is used (the only exception to this occurs when $j = 0$). It follows that not all the digital codes are used either, in fact only the upper part of the available digital codes is used.

To guarantee a monotonic transition between the codes generated by two adjacent integration slots, the first digital code generated by slot $j+1$ has to be at least equal to the last digital code generated by slot j.

From Equation 2.14 and Figure 3-4:

$$\overbrace{E_{j+1} \cdot q\left(\frac{S_{MAX}}{T_j} \cdot T_{j+1}\right)}^{\text{First code of slot } j+1} \geq \overbrace{E_j \cdot q\left(\frac{S_{MAX}}{T_j} \cdot T_j\right)}^{\text{Last code of slot } j} \qquad (3.3)$$

$$\frac{T_0}{T_{j+1}} \cdot \left\lfloor 2^N \cdot \frac{T_{j+1}}{T_j} \right\rfloor \geq \frac{T_0}{T_j} \cdot \left(2^N - 1\right) \qquad (3.4)$$

If for $M > 1$, if an integration slot ratio set is defined as:

$$\mathcal{R} = \{R_0, R_1, \ldots, R_{M-2}\} = \left\{\frac{T_0}{T_1}, \frac{T_1}{T_2}, \ldots \frac{T_{M-2}}{T_{M-1}}\right\} \qquad (3.5)$$

Then to have a monotonic sensor transfer characteristic:

$$C\left(N, \mathcal{R}\right) = R_j \cdot \frac{\left\lfloor \frac{2^N}{R_j} \right\rfloor}{2^N - 1} \geq 1 \quad , \quad \forall \; R_j \in \mathcal{R} \qquad (3.6)$$

When $C\left(N, \mathcal{R}\right) = 0$ the slot ratio is too extreme for the available ADC resolution and the normalized illumination (I/I_{REF}) in the range $\left[E_{j+1}, E_{j+1} \cdot \left(1 + 1/2^N\right)\right]$ produces a pixel signal that falls in the first ADC bin (digital code 0, Figure 3-5(a)).

When $C\left(N, \mathcal{R}\right) = 1$ the last digital code generated by slot j is the same as the first digital code generated by slot $j+1$, so the last intensity bin of size $(E_j \cdot I_{REF})/2^N$ is lost and the transition point between sections of the transfer characteristic that are scaled by E_j and E_{j+1} is shifted, resulting in a bin of size of $\left(I_{REF}/2^N\right) \cdot (E_j + E_{j+1})$ (Figure 3-5(c)).

Figure 3-6 shows the allowable ADC resolution–integer slot ratio combinations. It is clear that Equation 3.6 severely limits the converter resolutions for some particular ratios, but for mid– and high–resolution ADCs all ratios of the form $R = 2^a$ for some integer $a \leq N$ are available and satisfy $C\left(N, \mathcal{R}\right) > 1$:

$$C\left(N, \mathcal{R}\right) = 2^a \cdot \frac{\left\lfloor \frac{2^N}{2^a} \right\rfloor}{2^N - 1} = \frac{2^N}{2^N - 1} \geq 1 \text{ as } \left\lfloor \frac{2^N}{2^a} \right\rfloor = \frac{2^N}{2^a} \text{ when } a \leq N \qquad (3.7)$$

The integration slot ratios need not be integer, and in practice, due to the tolerances inherent in any hardware system, will hardly ever be. Figure 3-5(e) and Figure 3-5(f) show two transfer characteristic examples for non–integer integration slot ratios. In the first case

(a) $C(N, \mathcal{R}) = 0$, $N = 3$, $\mathcal{T} = \{1, 1/16\}$.

(b) $0 < C(N, \mathcal{R}) < 1$, $N = 3$, $\mathcal{T} = \{1, 1/5\}$.

(c) $C(N, \mathcal{R}) = 1$, $N = 3$, $\mathcal{T} = \{1, 1/7\}$.

(d) $C(N, \mathcal{R}) > 1$, $N = 3$, $\mathcal{T} = \{1, 1/8\}$.

(e) $C(N, \mathcal{R}) > 1$, $N = 3$, $\mathcal{T} = \{1, 1/3.7\}$.

(f) $0 < C(N, \mathcal{R}) < 1$, $N = 3$, $\mathcal{T} = \{1, 1/4.4\}$.
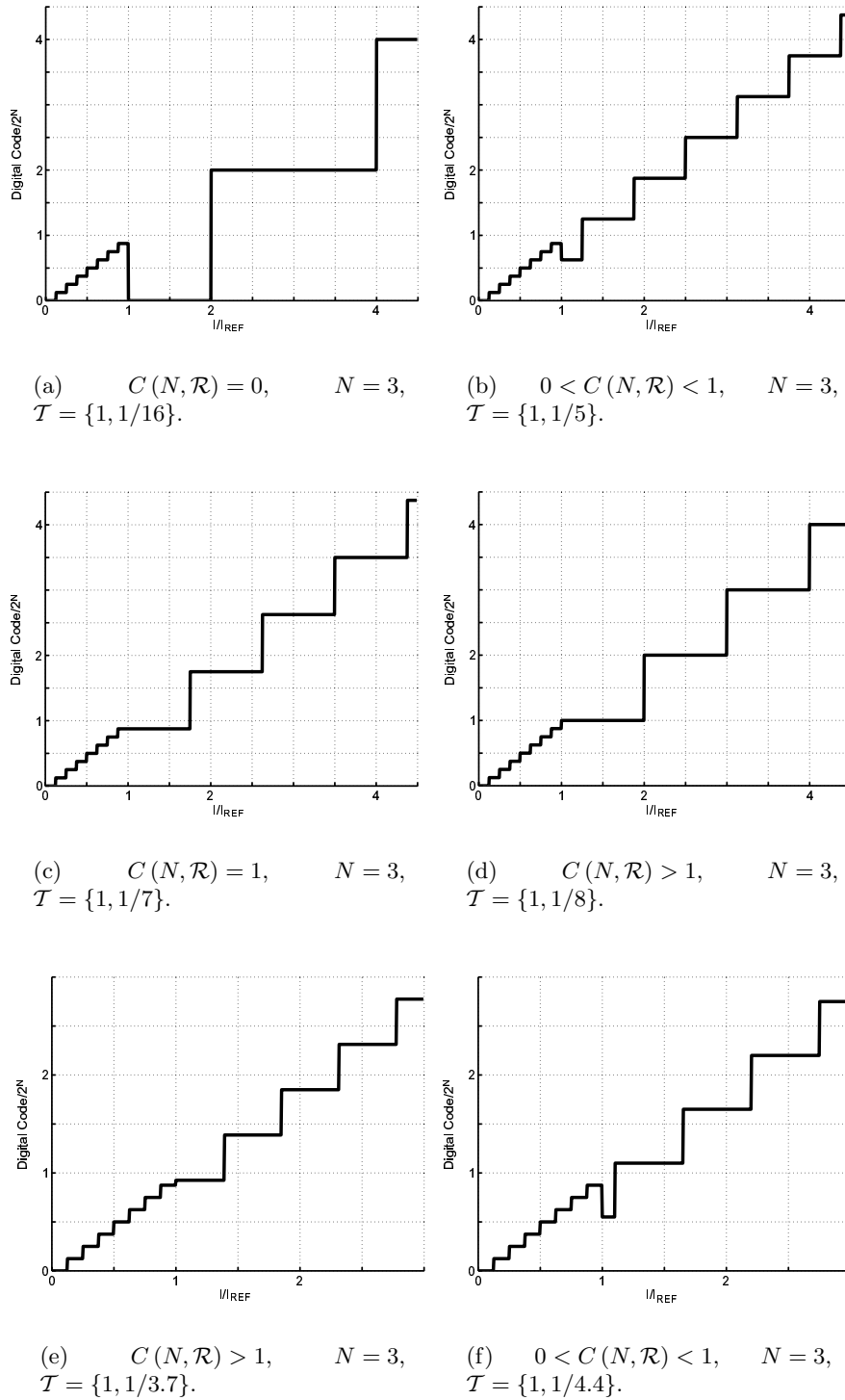
Figure 3-5: Sample image sensor transfer characteristics for different analog–to–digital converter resolution (N)–integration slot ratios (R) combinations.
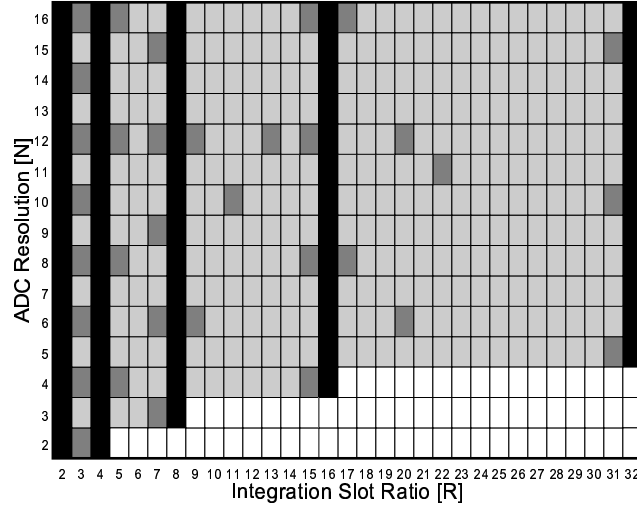
Figure 3-6: Allowable analog–to–digital converter resolution (N)–integer integration slots ratio (R) combinations. $C(N, \mathcal{R}) = 0$ denoted by white, $0 < C(N, \mathcal{R}) < 1$ denoted by light gray, $C(N, \mathcal{R}) = 1$ denoted by dark gray and $C(N, \mathcal{R}) > 1$ denoted by black. The sensor transfer characteristic is monotonic only when $C(N, \mathcal{R}) \geq 1$.

the curve is monotonic but the first digital code step size of the section scaled by E_{j+1} is significantly smaller than the rest. In this case the transfer characteristic is not monotonic as $C(N, \mathcal{R}) < 1$.

3.4.2   Pixels with non–destructive read and conditional reset capabilities

Pixels with non–destructive read capability are necessary to select the optimal integration slot without introducing non–linearities in the pixel signal. The pixels also need to have a conditional reset feature because in general the intensity changes from frame to frame so a single pixel may need to be reset at different points in time during different integration cycles.

3.4.3   Storage

A memory is needed because the selection of the optimal integration slot takes place during the integration time, but the total pixel output calculation occurs after the integration time has ended. Therefore the integration slot used needs to be stored on a per–pixel basis until the pixel quantization is done. The size in bits (B) of this per–pixel storage

element is given by:

$$B = \lceil log_2 |\mathcal{E}| \rceil = \lceil log_2 M \rceil \tag{3.8}$$

where $\lceil \cdot \rceil$ denotes the ceiling function. The contents of this memory can be indexes to an exposure ratio look–up table or they can represent exponents if the integration slot ratio set is of the form $\mathcal{R} = \{R, \ldots, R\}$ so that the exposure ratio set is $\mathcal{E} = \{R^0, \ldots, R^{M-1}\}$ for some integer $R > 0$.

The memory contents (b) have to be zeroed at the start of the integration time and are accessed twice per pixel check j. They are first read to determine if the pixel was reset in the previous check: if $b < j - 1$ then the pixel was reset before the last pixel check and does not need to be reset again. If $b = j - 1$ then the pixel was reset during the last check cycle and the predictive decision has to be made once again. If it is determined that the pixel is going to saturate before the end of the integration time the pixel needs to be reset and the memory contents have to be updated with $b = j$. At the end of the integration time the memory contents will be the optimal integration slot index i, that is, i is equal to the first check j in which it was determined the pixel was not going to saturate. Since the memory is accessed twice during each pixel check, the location and performance of this per–pixel memory directly affects the dynamic range increase E_{MAX} (Section 3.5.5).

3.4.4 Integration controller

An integration controller is necessary to implement the decision process involved in each check of the dynamic range expansion algorithm. Its location and performance also affect the dynamic range increase E_{MAX}. This subsystem needs to access the pixel, compare its signal with the signal threshold, reset the pixel when necessary and update the associated pixel memory contents.

From an implementation standpoint it is simpler to have $R_j = R \; \forall \; j \in \{0, \ldots, M - 2\}$ because then the comparison thresholds $S_{TH}^{T_j}$, $j \in \{1, \ldots, M - 1\}$ are constant:

$$S_{TH}^{T_j}(t_{j+1}) = \frac{S_{MAX}}{T_j} \cdot (T_j - T_{j+1}) = S_{MAX} \cdot \left(1 - \frac{1}{R}\right) \tag{3.9}$$

Further system simplification is achieved if $R = 2^a$ for some integer $a > 0$ since in this case a dedicated arithmetic unit is not required to produce the total pixel signal, $S_{TOT}(T_i)$ can be obtained by simply shifting the quantized pixel signal $q(S(T_i))$ according to E_i.
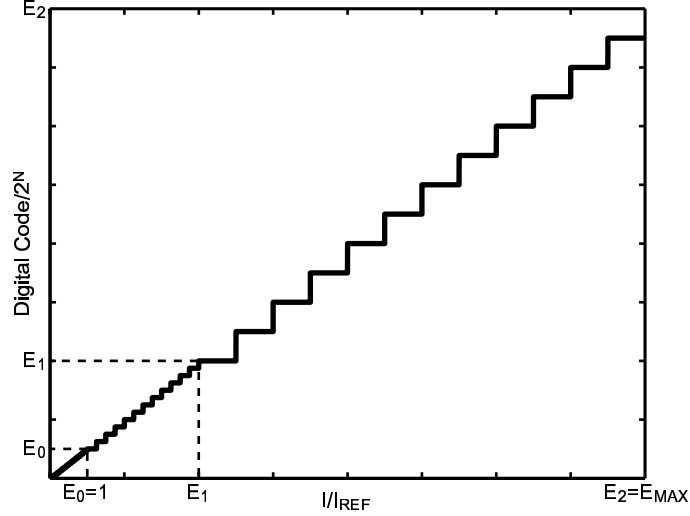
Figure 3-7: Sample transfer characteristic for an image sensor that implements the predictive multiple sampling algorithm. $\mathcal{T} = \{T_0, T_0/4, T_0/16\}$ and a 4–bit analog–to–digital converter with an input range of S_{MAX} used. Intensity normalized to $I_{REF} = I_{TH}(T_{INT})$, the intensity threshold for an integration time T_{INT}. Digital code normalized to 2^N.

3.5 Performance

3.5.1 Transfer Characteristic

The algorithm adaptively pre–scales the intensity received to try to maintain the pixel signal in its linear range. With $S \approx K_1 \cdot I \cdot t$ then the quantized pixel signal after integration slot j has ended is:

$$\begin{aligned} S_q(T_j) &= \left\lfloor \frac{2^N \cdot S(T_j)}{S_{MAX}} \right\rfloor = \left\lfloor \frac{2^N \cdot K_1 \cdot I \cdot T_j}{K_1 \cdot I_{TH}(T_{INT}) \cdot T_{INT}} \right\rfloor \\ &= \left\lfloor 2^N \cdot \frac{I}{I_{REF}} \cdot \frac{T_j}{T_0} \right\rfloor = \left\lfloor \frac{2^N}{E_j} \cdot \frac{I}{I_{REF}} \right\rfloor \end{aligned} \tag{3.10}$$

where $\lfloor \cdot \rfloor$ denotes the floor function, $T_{INT} = T_0$, $I_{REF} = I_{TH}(T_{INT})$ and S_{MAX} is also assumed to be the input range of the ADC. The intensity bin size of the resulting quantizer is therefore:

$$\Delta I_j = \frac{E_j \cdot I_{REF}}{2^N} \ , j \in \{0, 1, \dots, M-1\} \tag{3.11}$$

The intensity bins (and thus the intensity quantization noise) are integration slot–dependent and increase from a minimum of $I_{REF}/2^N$ to a maximum of $E_{MAX} \cdot I_{REF}/2^N$ (Figure 3-7).

The total pixel signal (Equation 2.14) can be re–written as:

$$S_{TOT}\left(T_i\right) = E_i \cdot q\left(\frac{1}{E_i} \cdot \frac{I}{I_{REF}}\right) \ , i \in \{0, 1, \ldots, M-1\} \tag{3.12}$$

It can be seen that the intensity is scaled down in the analog domain to meet the bounded pixel dynamic range. The scaling is subsequently undone in the digital domain, where there are more flexible dynamic range restrictions.

3.5.2 Signal–to–Noise Ratio

The signal–to–noise ratio (SNR) depends on the integration time and intensity received. The total photo–generated charge Q_{PH} collected in T_{INT} for a pixel with a photodiode area A_{PH} and quantum efficiency $\eta\left(\lambda\right)$ that receives an optical power per unit area I at a wavelength λ is:

$$Q_{PH}\left(I, T_{INT}\right) = \frac{q \cdot A_{PH} \cdot \lambda \cdot \eta\left(\lambda\right)}{h \cdot c} \cdot I \cdot T_{INT} = K_2 \cdot I \cdot T_{INT} \tag{3.13}$$

where $h = 6.624 \cdot 10^{-34}$ J $\cdot$ s (Planck's constant) and $c = 3 \cdot 10^8$ m/s (speed of light). The photon arrival process can be modeled as a Poisson process so the photon uncertainty (standard deviation) from the average is;

$$\sigma_{Q_{PH}}\left(I, T_{INT}\right) = \sqrt{Q_{PH}\left(I, T_{INT}\right)} \tag{3.14}$$

Consequently the photon shot noise–limited SNR is:

$$SNR\left(T_{INT}\right) = \frac{Q_{PH}\left(T_{INT}\right)}{\sigma_{Q_{PH}}\left(T_{INT}\right)} = \sqrt{K_2 \cdot I \cdot T_{INT}} \tag{3.15}$$

As the transfer characteristic, the SNR of the image sensor can be divided in regions depending on which integration slot was used. The maximum SNR is achieved at the end of each integration slot:

$$SNR_{MAX} = \sqrt{K_2 \cdot I_{TH}\left(T_j\right) \cdot T_j} \tag{3.16}$$

With $T_j = T_0/E_j$, $I_{TH}\left(T_j\right) = E_j \cdot I_{TH}\left(T_0\right)$, $T_{INT} = T_0$ and $I_{REF} = I_{TH}\left(T_{INT}\right)$:

$$SNR_{MAX} = \sqrt{K_2 \cdot I_{REF} \cdot T_{INT}} \tag{3.17}$$
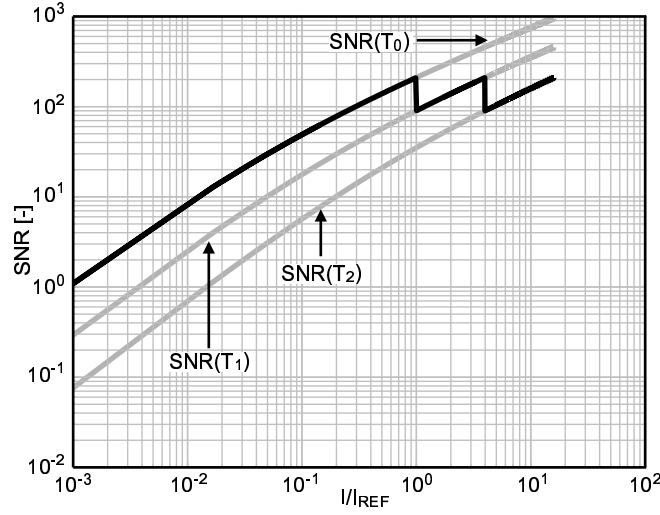
Figure 3-8: Sample signal–to–noise ratio for an image sensor that implements the predictive multiple sampling algorithm. $\mathcal{T} = \{T_0, T_0/4, T_0/16\}$ and $I_{REF} = I_{TH}(T_{INT})$ used.

then, using Equation 3.5, the SNR reduction at the integration slot transitions is:

$$SNR_{MAX}(T_j) = \sqrt{R_j} \cdot SNR_{MIN}(T_{j+1}) \tag{3.18}$$

The preceding derivation assumes that the SNR is in the photon shot noise–limited region. If this is not the case the SNR drop at the slot transitions is bigger than calculated, its exact value depending on the pixel noise floor. Figure 3-8 shows the SNR for the sample image sensor with $\mathcal{T} = \{T_0, T_0/4, T_0/16\}$ whose transfer characteristic is shown in Figure 3-7.

Slot 0 uses the original pixel SNR (the maximum available) so the low light performance of the image sensor is not affected by the predictive multiple sampling algorithm.

3.5.3 Exposure Control

Per–pixel or time–shared integration controllers can extend the predictive multiple sampling algorithm to every pixel of a sensing array. This provides full–frame adaptive exposure control: integration slots are automatically selected for each pixel according to the intensity they receive, and at any point in time multiple integration slots can be used concurrently throughout the array.

All the elements that need to be added to implement the predictive multiple sampling algorithm (integration controller and memory) can be integrated on a single–chip hardware

solution along with the pixel array and data conversion block. From a computational perspective, the predictive algorithm does almost all of its work during the integration time, and the only operation that needs to be done after the pixel quantization, the total pixel signal calculation (Equation 2.14), is purely combinational and thus can be performed at high speed. Consequently, from a user perspective, an image snesor implementing the predictive multiple sampling algorithm behaves and responds as any other image sensor but with a higher dynamic range.

3.5.4   Light Intensity Dynamic Range Increase

The intensity dynamic range increase provided by the multiple sampling algorithm (E_{MAX}) over the photodiode dynamic range is directly proportional to the integration time T_{INT} and inversely proportional to the shortest integration slot T_{M-1} (Equation 2.12). The integration time is typically upper bounded by system–level factors such as a desired frame rate, minimization of errors due to incorrect predictions, motion–induced blur minimization, etc. The shortest integration slot T_{M-1}, on the other hand is typically implementation–dependent and therefore difficult to bound in a generalized case.

A pixel that includes the integration controller and ADC is the highest performing implementation of the multiple sampling algorithm. In this case the shortest integration slot is only limited by how accurately its length can be controlled. However, with current fabrication technologies said functionality in the pixel implies a large pixel area for any practical fill factor which either severely limits the sensor spatial resolution or increases cost [55]. The on–pixel memory sometimes is implemented in the analog domain [51, 56], which also presents problems due to the extra quantization required to obtain E_i, the potential corruption of the stored value due to crosstalk between the photodiode and the storage node and the pixel area increase.

Another alternative is to time–share the integration controller. Here if a single pixel check takes T_C seconds to complete and PPC (**p**ixels **p**er **c**ontroller) pixels time–share an integration controller, it takes $T_{C_{TOT}} = PPC \cdot T_C$ seconds for the controller to check all of its pixels and be ready for another check cycle. Therefore:

$$T_{M-2} - T_{M-1} \geq T_{C_{TOT}} \Longrightarrow T_{M-1} \geq \frac{PPC \cdot T_C}{R_{M-2} - 1} \qquad (3.19)$$

Time–sharing the controller limits the minimum integration slot and by extension E_{MAX}. Lowering this bound can be achieved by minimizing PPC and/or T_C. PPC can be minimized by placing the controllers in the sensing array, sharing one with a small neighborhood of pixels. This often restricts the comparator architecture options and results in challenging layouts in order to achieve reasonable fill factors and maintain the distance between the geometric centers of the photodiodes constant [57]. A compromise solution is to have column–parallel integration controllers [58].

The timing of a pixel check highlights the options available for minimizing T_C:

1. Read pixel memory (T_{RM} seconds) and pixel signal (T_{PR} seconds).

2. When data ready perform comparison in T_{COMP} seconds (only necessary if pixel was reset in previous check cycle).

3. If necessary, write pixel memory (T_{WM} seconds) and reset pixel (T_{PRST} seconds).

So the total check time is:

$$T_C = max\left(T_{RM}, T_{PR}\right) + T_{COMP} + max\left(T_{WM}, T_{PRST}\right) \tag{3.20}$$

Both memory access time (T_{RM}, T_{WM}) and system power dissipation increase substantially if the memory is not on the same die as the image sensor, so there is a trade–off between sensor dynamic range and die area/cost.

For large spatial resolution arrays with on–chip memory it is possible to have $T_{RM} \ll T_{PR}$ and $T_{WM} \ll T_{PRST}$, that is, a scenario where pixel access dominates the check time T_C. Both T_{PR} and T_{PRST} in turn are dominated by parasitics that scale with the number of pixels per controller (PPC) so the same trade–offs present in the PPC minimization apply to this case.

Non–negligible pixel check times ($T_C \gg 0$) split the actual pixel check time and the potential start of the following integration slot. However, under the constant illumination assumption there is a linear relationship between time and comparison threshold $S_{TH}^{T_j}(t)$ (Equation 3.1), so the start of the integration slots remain at $t_j, j \in \{1, \cdots, M-1\}$ but the pixel checks start at $t_j - T_C$ with a comparison threshold of $S_{TH}^{T_{j-1}}(t_j - T_C)$.
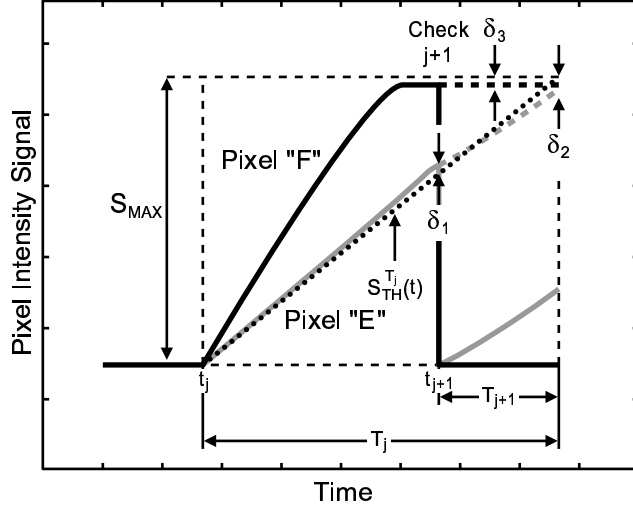
Figure 3-9: Predictive multiple sampling algorithm behavior for two pixels that receive decreasing intensity during the integration time. An unnecessary pixel reset leads to a suboptimal (shorter) integration slot usage.

3.5.5   Optimality of the Integration Slot Selection

The intensity quantization noise increases as shorter integration slots are used (Equation 3.11). Consequently it is desirable to use the longest integration slot that does not saturate the pixel in order to avoid losing visual information (edges, textures, etc.) due to a coarser quantization.

In some instances optimality is not achieved because the constant intensity condition is not fully satisfied. Figure 3-9 exemplifies the case when the optimal integration slot is slot j, but the temporal evolution of the light intensity places the pixel signals above the threshold at the pixel check, so slot $j+1$ is used instead to produce the total pixel signal. The error (ΔS_{TOT}) measured in number of digital codes produced by this unnecessary reset cycle is bounded by:

$$E_j \cdot \left(2^N - 1\right) \geq \Delta S_{TOT} \begin{array}{ll} \geq & 0 \quad , \; C\left(N, R\right) = 1 \\ > & E_{j+1} \quad , \; C\left(N, R\right) > 1 \end{array} \tag{3.21}$$

The lower bound is approached when the unchecked $S\left(t\right)$ is close to $S_{TH}^{T_j}\left(t\right)$, specifically when $S\left(t_{j+1}\right) = S_{TH}^{T_j}\left(t_{j+1}\right) + \delta_1{}^3$ and also $S\left(T_j\right) = S_{MAX} - \delta_2$ (Pixel "E" in Figure 3-9).

---

[3] $\delta_x \to 0, x \in \{1, \cdots, 3\}$.
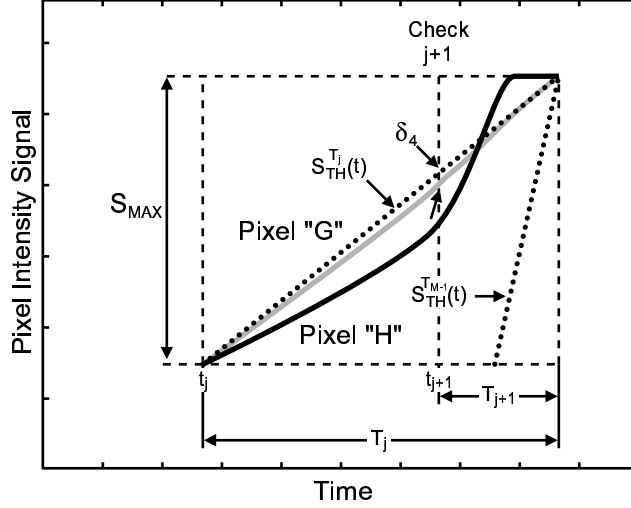
Figure 3-10: Predictive multiple sampling algorithm behavior for two pixels that receive increasing intensity during the integration time. Pixels are not reset which leads to a longer integration slot usage.

The upper bound of the inequality is reached when $S(t_{j+1}) = S(T_{j+1}) = S_{MAX} - \delta_3$ (Pixel "F" in Figure 3-9).

When the pixels receive an increased intensity during the integration time but their signals are below the threshold at some check j the pixels are not reset and slot j is used when a shorter slot would have been optimal (Figure 3-10). Therefore the pixels saturate before the end of slot j. In this case the error is bounded by:

$$(E_{MAX} - E_j) \cdot (2^N - 1) \geq \Delta S_{TOT} \geq 0 \tag{3.22}$$

The lower bound of the inequality is again approached when the unchecked $S(t)$ is close to $S_{TH}^{T_j}(t)$. The upper bound of the inequality is reached when the intensity received after the $j+1$ pixel check saturates the photodiode even when the shortest integration slot $M-1$ is used. In both cases the error due to the incorrect predictions is reduced if the integration time $T_0 = T_{INT}$ is itself reduced. However, this also lowers the maximum dynamic range increase E_{MAX} (Equation 2.12).

Incorrect predictive decisions can also be made if the comparison against the threshold is done in the analog domain and the analog comparator has a finite but unknown offset (C_{OS}). A solution to this problem is to set the reset threshold level to $S_{TH}^{T_j}(t_{j+1}) - |C_{OS}|$.

Pixels whose signals at the check $S(t_{j+1})$ lie in the range $\left(S_{TH}^{T_j}(t_{j+1}), S_{TH}^{T_j}(t_{j+1}) - |C_{OS}|\right)$ are reset when ideally they should not be, so the integration slot used for them is suboptimal. However, silicon area permitting some well–known offset reduction techniques can be used to make $C_{OS} \ll S_{MAX}$ so the intensity range that is suboptimally quantized is narrow.

3.6   Selection of Optimal Integration Slot Set of Given Size

The optimal integration slot for a particular pixel receiving a given illumination is the longest slot that does not saturate the pixel as this ensures that the largest possible SNR will be achieved and the smallest possible light intensity–to–digital code (I2D) quantization bin will be used. However, in a sensing array several pixels can receive vastly different illuminations so with a finite integration slot set this degree of optimality in general cannot be achieved for every pixel. Consequently, given an integration slot set size $|\mathcal{T}|$, the optimal integration slot set $\mathcal{T}_{OPT}$, $|\mathcal{T}_{OPT}| = |\mathcal{T}|$ for a particular scene is the one that minimizes the average I2D quantization noise.

3.6.1   Derivation

The I2D quantization noise ΔI is the difference between a particular illumination level I and the reconstruction point produced by the sensor for I[4] (Figure 3-11). As Figure 3-12 shows, the I2D quantization error lies, for every section of the transfer characeristic, between $-\Delta I_k/2$ and $\Delta I_k/2$. Formally, using Equation 3.10, the expected value of the I2D quantization noise $E\left[\Delta I^2\right]$ is then:

$$E\left[\Delta I^2\right] = \sum_{k=0}^{M-1} \int_{I_{REF} \cdot E_{k-1}}^{I_{REF} \cdot E_k} \Delta I^2 \cdot p_I(I) \cdot dI \tag{3.23}$$

$$= \sum_{k=0}^{M-1} \int_{I_{REF} \cdot E_{k-1}}^{I_{REF} \cdot E_k} \left(I - \left(\frac{1}{2} + \left\lfloor \frac{2^N}{E_k} \cdot \frac{I}{I_{REF}} \right\rfloor\right) \cdot \Delta I_k\right)^2 \cdot p_I(I) \cdot dI \tag{3.24}$$

$$\exists \, \mathcal{T}_{OPT}, \ |\mathcal{T}_{OPT}| = M \ : \ E\left[\Delta I^2\right]_{\mathcal{T}_{OPT}} \leq E\left[\Delta I^2\right]_{\mathcal{T}} \ \forall \ |\mathcal{T}| = M \tag{3.25}$$

where $\Delta I_k = E_k \cdot I_{REF}/2^N$ (Equation 3.11), $E_{-1} = 0$, and $p_I(I)$ is the probability density function of the illumination received by the sensor in a given frame, i.e. the probability that a given pixel will receive an illumination I.

---

[4]The chosen reconstruction points bisect the bins.
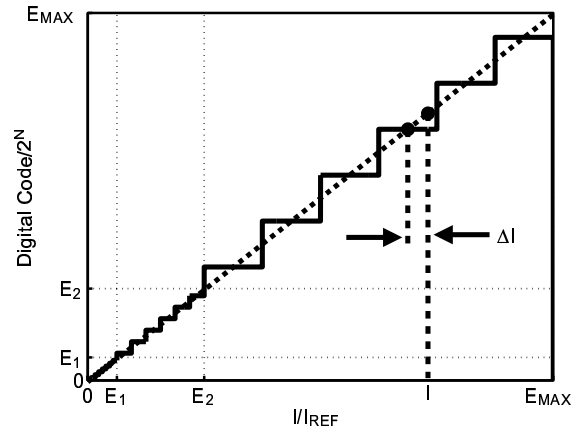
Figure 3-11: ample transfer characteristics showing the intensity–to–digital code quantization noise for a particular illumination. $\mathcal{T} = \{T_0, T_0/4, T_0/16\}$ and $N = 3$ used.
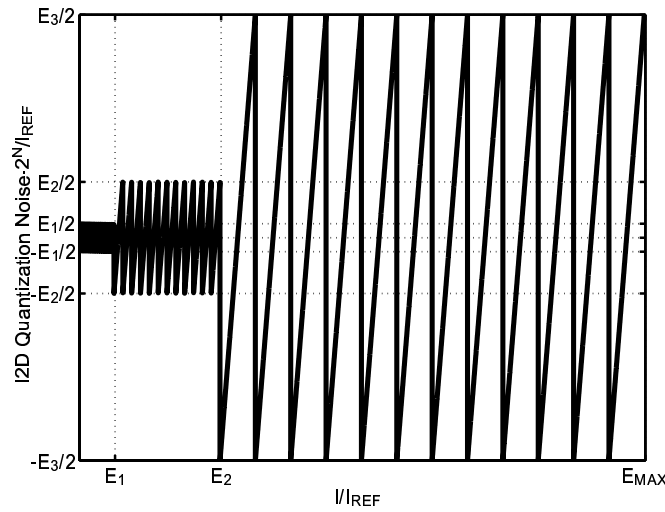
Figure 3-12: Light intensity–digital code quantization noise for sample transfer characteristic. $\mathcal{T} = \{T_0, T_0/4, T_0/16\}$ and $N = 3$ used.

Equation 3.24 cannot be implemented in hardware to determine the optimal integration slot of a give size since the actual illumination cannot be obtained with infinite precision through the inherent quantization operation of the image sensor. However, relevant information can still be obtained by calculating an upper bound of $E\left[\Delta I^2\right]$, assuming that
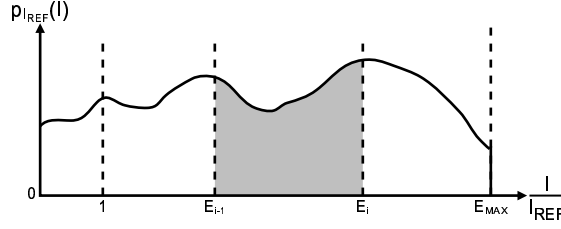
Figure 3-13: Sample probability density function of a natural scene. The illumination is normalized to the reference illumination I_{REF}.

$\Delta I = \Delta I_k$ for every illumination point[5]:

$$E\left[\Delta I^2\right] \leq E\left[\Delta I^2\right]_{UB} = \sum_{k=0}^{M-1} \int_{I_{REF}\cdot E_{k-1}}^{I_{REF}\cdot E_k} \Delta I_k^2 \cdot p_I\left(I\right) \cdot dI \tag{3.26}$$

$$= \sum_{k=0}^{M-1} \int_{I_{REF}\cdot E_{k-1}}^{I_{REF}\cdot E_k} \left(\frac{E_k \cdot I_{REF}}{2^N}\right)^2 \cdot p_I\left(I\right) \cdot dI \tag{3.27}$$

$$= \left(\frac{I_{REF}}{2^N}\right)^2 \cdot \sum_{k=0}^{M-1} E_k^2 \cdot \int_{I_{REF}\cdot E_{k-1}}^{I_{REF}\cdot E_k} p_I\left(I\right) \cdot dI \tag{3.28}$$

Further simplification of this expression can be achieved by noting that the minimum of $E\left[\Delta I^2\right]_{UB}$ is also the minimum of $E\left[\Delta I^2\right]_{UB} / \left(\frac{I_{REF}}{2^N}\right)^2$, and using the probability mass function of the normalized illumination I/I_{REF}, $p_{I_{REF}}\left(I\right)$ (Figure 3-13):

$$E\left[\Delta I^2\right]_{UB}' = \sum_{k=0}^{M-1} E_k^2 \cdot \int_{E_{k-1}}^{E_k} p_{I_{REF}}\left(I\right) \cdot dI \tag{3.29}$$

Typically the integration time $T_{INT} = T_0$ is fixed by system factors, so for a given scene its contribution to the quantization noise is constant and thus does not affect the location of its minimum. What is more, it is not necessary to compute Equation 3.29 for all possible real values of integration slot ratios in the range $[1, E_{MAX}]$, because Equation 3.6 limits these ratios to powers of 2 to achieve monotonic sensor transfer characteristics with constant digital code steps within its sections. Consequently the expression whose minimum needs to be found to determine the optimal integration slot set of size M is:

$$E\left[\Delta I^2\right]_{UB}'' = \sum_{k=1}^{M-1} E_k^2 \cdot \int_{E_{k-1}}^{E_k} p_{I_{REF}}\left(I\right) \cdot dI \tag{3.30}$$

---

[5]Any fraction of the intensity bin size ΔI_k would give the same location for the minimum of $E\left[\Delta I^2\right]_{UB}$.

$$E_1 < \ldots < E_{M-1}, \ E_j = 2^a \ , \ a \in \{1, 2, \ldots, log_2\left(E_{MAX}\right)\}, \ j \in \{1, \ldots M-1\} \quad (3.31)$$

Finally, even more computation savings can be achieved by noting that there are only $log_2\left(E_{MAX}\right)$ integrals that need to be calculated, namely:

$$INT\left(j\right) = \int_{2^{j-1}}^{2^j} p_{I_{REF}}\left(I\right) \cdot dI, \ j \in \{1, 2, \ldots, log_2\left(E_{MAX}\right)\} \quad (3.32)$$

so that:

$$\int_{2^a}^{2^b} p_{I_{REF}}\left(I\right) \cdot dI = \sum_{j=a+1}^{b} INT\left(j\right) \quad (3.33)$$

for integers $a, b \in \{1, 2, \ldots, log_2\left(E_{MAX}\right)\}$ and $a < b$. Therefore:

$$E\left[\Delta I^2\right]''_{UB} = \sum_{k=1}^{M-1} E_k^2 \cdot \sum_{j=log_2(E_{k-1})}^{log_2(E_k)} INT\left(j\right) \quad (3.34)$$

$$E_1 < \ldots < E_{M-1}, \ E_j = 2^a \ , \ a \in \{1, 2, \ldots, log_2\left(E_{MAX}\right)\}, \ j \in \{1, \ldots M-1\} \quad (3.35)$$

The restriction to the integration slot ratios imposed by Equation 3.6 makes it feasible to perform an exhaustive evaluation of Equation 3.34 to find the optimal integration slot set. It can be shown that for a given a maximum exposure ratio E_{MAX} and integration slot set size $|\mathcal{T}| \leq log_2\left(E_{MAX}\right) + 1$, the number of points to be computed is:

$$NP_{E[\Delta I^2]''_{UB}} = \begin{pmatrix} log_2\left(E_{MAX}\right) - 1 \\ |\mathcal{T}| - 2 \end{pmatrix} = \frac{\left(log_2\left(E_{MAX}\right) - 1\right)!}{\left(|\mathcal{T}| - 2\right)! \cdot \left(log_2\left(E_{MAX}\right) - |\mathcal{T}| - 3\right)!} \quad (3.36)$$

where ! denotes the factorial operation. Figure 3-14 shows the number of data points that need to be computed for all possible integration slot set sizes given a particular maximum exposure ratio E_{MAX}. As it can be seen, less than 500 points are needed to find the optimal integration slot set, even when $E_{MAX} = 4096$ which adds 72dB to the original pixel illumination dynamic range.

3.6.2 Procedure

The optimal integration slot set for different set lengths can be found as follows:

- $\underline{|\mathcal{T}| = 1}$: $\mathcal{T} = \{T_0\} = \{T_{INT}\}$, which is usually determined by system factors such as desired frame rate, blur minimization, etc.
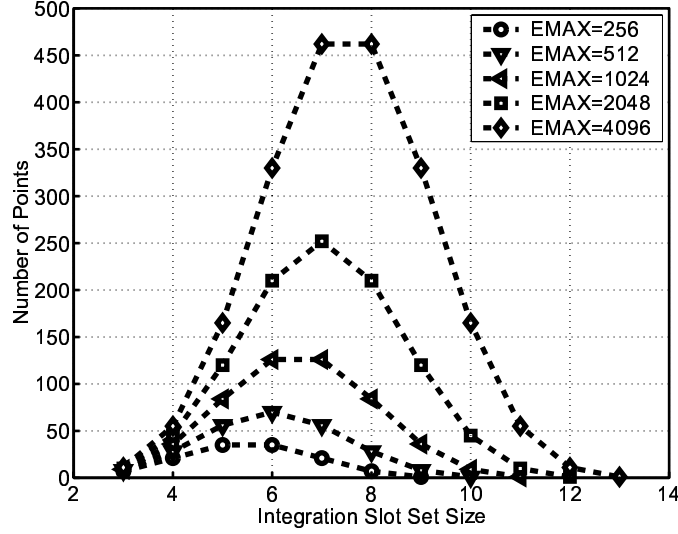
Figure 3-14: Number of data points that need to be computed to find the
minimum light intensity–digital code quantization noise for in-
tegration slot sets of different size.

- $\underline{|\mathcal{T}| = 2}$: $\mathcal{T} = \{T_0, T_1\}$. $T_0 = T_{INT}$, the first and longest integration slot is deter-
  mined as above. $T_1 = T_{LAST}$, the last integration slot, is determined by either the
  maximum illumination that is expected (I_{MAX}/I_{REF}) or the maximum illumination
  that the sensor can correctly acquire (limited by slot index storage capabilities, min-
  imum achievable integration slot, etc.). That is, the sensor has a fixed maximum
  $E_{MAX_{SYS}}$, so if $I_{MAX}/I_{REF} > E_{MAX_{SYS}}$ then $T_1 = T_{LAST} = T_{INT}/E_{MAX_{SYS}}$. But
  if the normalized illumination satisfies $I_{MAX}/I_{REF} < E_{MAX_{SYS}}$, then E_{MAX} has to
  be set as close to, but higher than, I_{MAX}/I_{REF} so as to capture all the illumination
  information using the longest possible integration slot. Mathematically:

$$E_{MAX} = 2^{\left\lceil log_2 \left(\frac{I_{MAX}}{I_{REF}} \right) \right\rceil} \tag{3.37}$$

- $\underline{|\mathcal{T}| = M}$: $\mathcal{T} = \{T_0, T_1, \ldots, T_{M-1}, T_{M-2}\}$. $T_0 = T_{INT}$ and $T_{M-1} = T_{LAST}$ are deter-
  mined as above. $T_1, \ldots, T_{M-2}$ are determined by first finding the integration slot ratios
  $E_1, \ldots, E_{M-1}$ that   minimize   Equation 3.34   and   then   operating:
  $T_1 = T_0/E_1, \ldots, T_{M-1} = T_0/E_{M-1}$.

3.6.3 Examples

Two scenes rendered with the Radiance synthetic imaging system [59] were used to empirically verify the proposed optimization method (Figures 3-15 (a) and (b)[6]). This software is able to produce wide dynamic range images with floating point output, making it ideal for simulation of natural scenes. In each case both the original wide dynamic range image and a brightness–equalized image (for a more uniform illumination PDF) were used as inputs to a Matlab [60] script that implements the procedure to find the optimal integration slot set as outlined above.

The simulated expected I2D quantization noise and the upper bound as computed by Equation 3.34 (normalized to their maximum values) were compared for $|\mathcal{T}| = 3$. Figure 3-15 shows that the length of the the second integration slot $T_1 = T_0/E_1$ that minimizes the expected quantization noise is correctly calculated by its upper bound. Tests for $|\mathcal{T}| > 3$ also resulted in correct selections of the optimal integration slot set. Care has to be taken so that the integration slot ratios do not exceed 2^N as mandated by Equation 3.6, a situation that becomes more likely for large E_{MAX} and small integration slot set sizes, depending on the image statistics.

Equation 3.34 becomes a better approximation as the ADC resolution increases because the sensor transfer characteristic bin sizes decrease exponentially as a function of N (Equation 3.11). Then:

$$2^N \gg 0 \implies \Delta I_j = \frac{E_j \cdot I_{REF}}{2^N} \longrightarrow \Delta I \ , j \in \{0, 1, \ldots, M - 1\} \tag{3.38}$$

That is to say, as the bin sizes decrease the quantization error becomes smaller and closer to the bin size. Figure 3-17 shows the upper bound and the exact error for different converter resolutions. As it can be seen, for medium and high resolutions the upper bound is not only feasible to compute but also extremely accurate.

3.6.4 Image Statistics Extraction

The calculation of the upper bound of the I2D quantization noise expected value assumes prior knowledge of the image statistics in the form of the scene illumination probability density function $p_{I_{REF}} (I/I_{REF})$. However, with the following procedure the image

---

[6]Obtained from public image gallery section of Radiance website, authors unknown.

(a) Office cubicle image.

(b) Drafting office image.
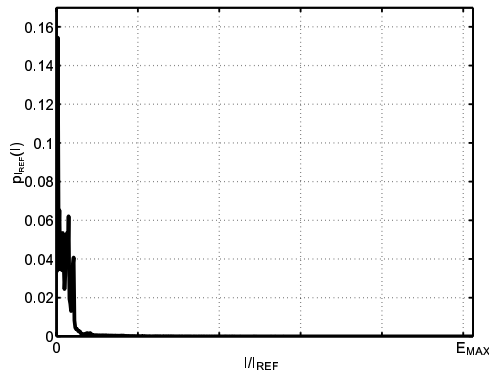
(c) PDF for the original office cubicle wide dynamic range image.
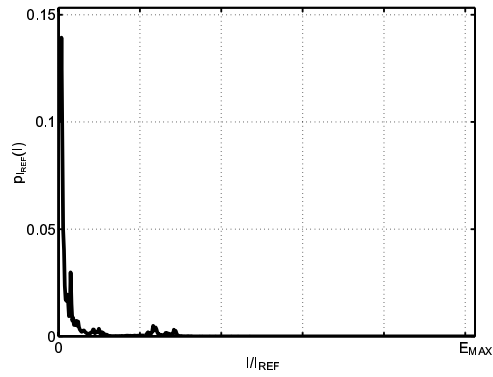
(d) PDF for the original draft office wide dynamic range image.
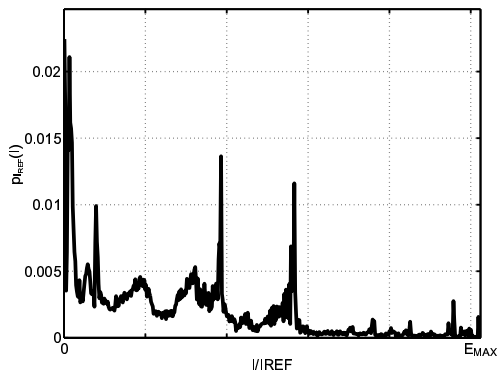
(e) PDF for the office cubicle brightness–equalized image.
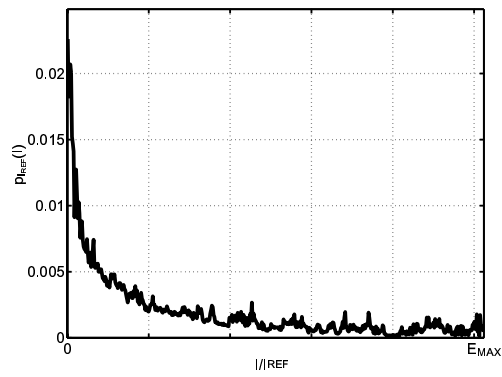
(f) PDF for the draft office brightness–equalized image.

Figure 3-15: Test scenes used to verify the proposed method to find the optimal integration slot set.
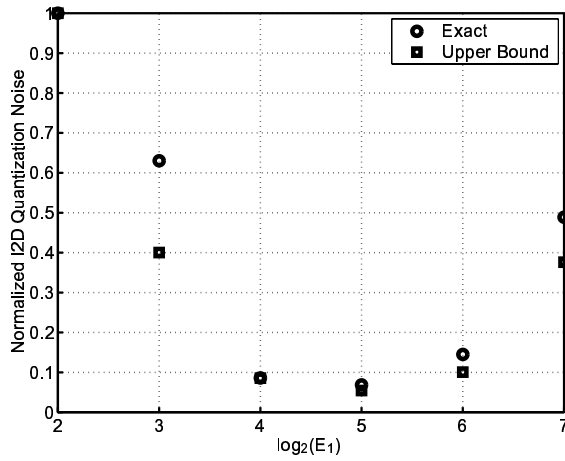
(a) Wide dynamic range office cubicle image. Minimum at $E_1 = 32$.
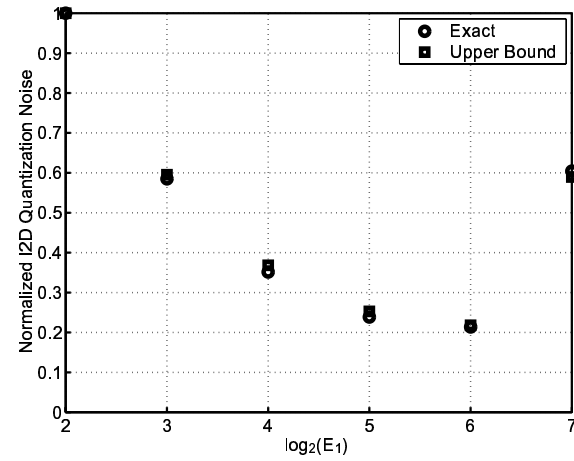
(b) Wide dynamic range draft office image. Minimum at $E_1 = 64$.
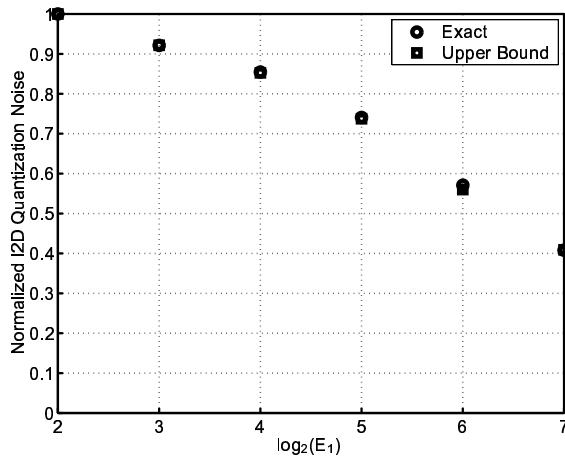
(c) Brightness–equalized office cubicle image. Minimum at $E_1 = 128$.
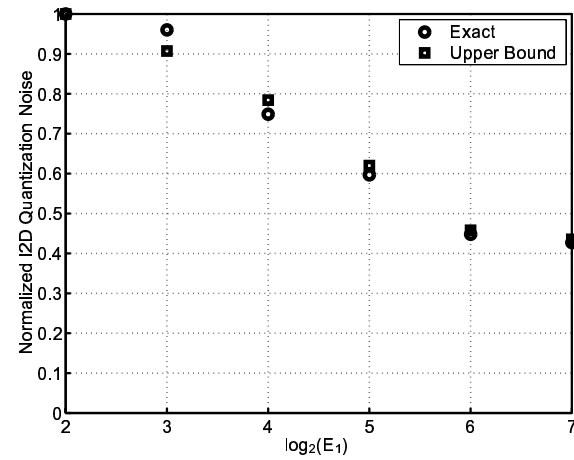
(d) Brightness–equalized draft office image. Minimum at $E_1 = 128$.

Figure 3-16: Comparison between the exact I2D quantization noise and the proposed approximation for the test scenes. $E_{MAX} = 256$ and $N = 4$ used. Curves normalized to their respective maximum values within the displayed range.

histogram can be used as a good approximation to these statistics for the purposes of the calculation of the optimal integration slot set:

1. Acquire a frame.

2. Calculate image histogram and appropriately scale it to obtain a valid probability
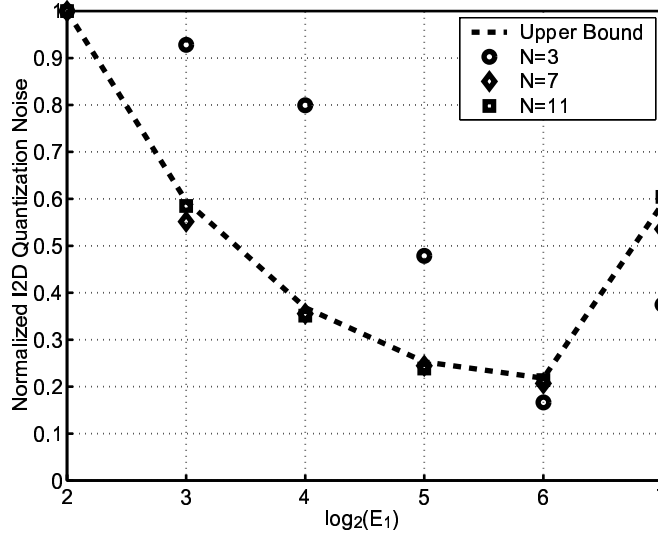
Figure 3-17: Comparison between the light intensity-to–digital code quanti-
zation noise and its upper bound for different analog–to–digital
converter resolutions. Wide dynamic range draft office scene
used with $E_{MAX} = 256$. Curves normalized to their respective
maximum values within the displayed range.

mass function.

3. Determine the optimal integration slot set of desired length $|\mathcal{T}|$ using Equation 3.34
   and the extracted PMF.

4. Repeat from step 1 till optimal integration slot sets from consecutive frames are iden-
   tical.

The draft office image was used as an example for this procedure, with $E_{MAX} = 32$,
$N = 4$ and $|\mathcal{T}| = 3$. Under these conditions, with the PDF shown in Figure 3-15 (d) the
optimal integration slot set is $\{T_0.T_0/8, T_0/32\}$. After the first frame the PMF shown in
Figures 3-18 (a) results, which gives an optimal integration slot set $\{T_0.T_0/16, T_0/32\}$. After
the second frame the PMF shown in Figures 3-18 (b) results, which gives an optimal inte-
gration slot set $\{T_0.T_0/8, T_0/32\}$. After the third frame the PMF shown in Figures 3-18 (c)
results, which gives the same optimal integration slot set as the second frame (which co-
incides with the optimal set calculated with the full–resolution PDF) and the procedure
stops. In general, larger slot set sizes and higher ADC resolutions produce more detailed
extracted PMFs and consequently reduce the number of iterations needed for the optimal
set to settle to its final value.

(a) PMF after first frame.

(b) PMF after second frame.

(c) PMF after third frame.
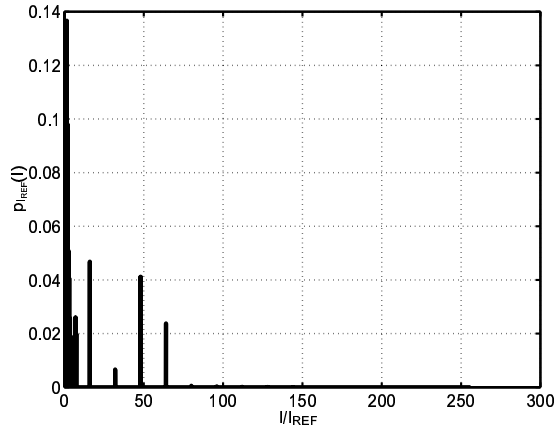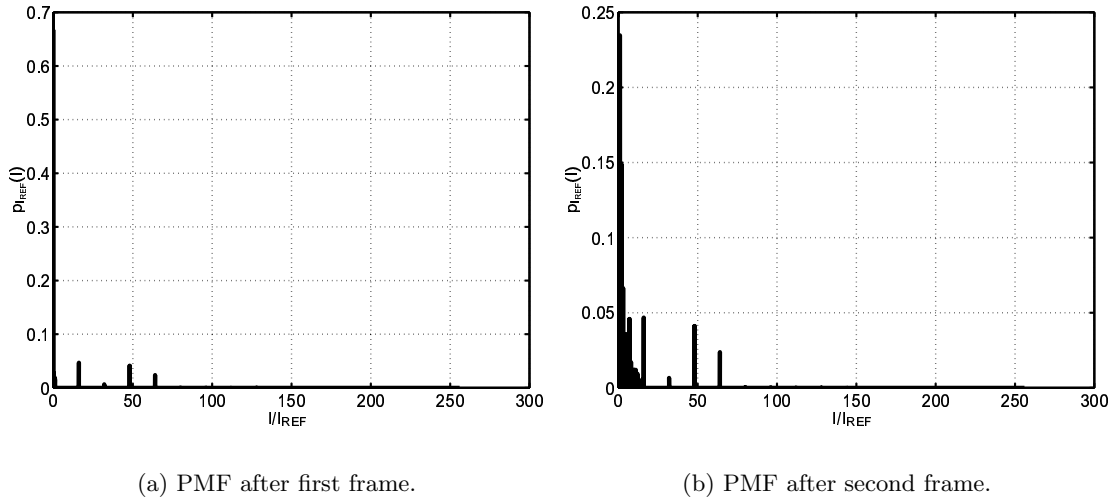
Figure 3-18: Extracted data used to approximate the image statistics needed in the optimal integration slot set determination.

3.7 Selection of the Integration Slot Set Size

There can be several valid integration slot sets that achieve a desired dynamic range expansion factor E_{MAX}. For instance, if a particular system needs $E_{MAX} = 256$, then $\mathcal{E}_1 = \{1, 1/2, 1/4, 1/8, 1/16, 1/32, 1/64, 1/128, 1/256\}$, $\qquad \mathcal{E}_2 = \{1, 1/4, 1/16, 1/64, 1/256\}$, $\mathcal{E}_3 = \{1, 1/16, 1/256\}$ and $\mathcal{E}_4 = \{1, 1/256\}$ are all possible options, using constant slot ratios R of 2, 4, 16 and 256 respectively. Which of the valid sets is the best depends on system constraints and the illumination statistics of the environment where the image sensor is

(a) Image.
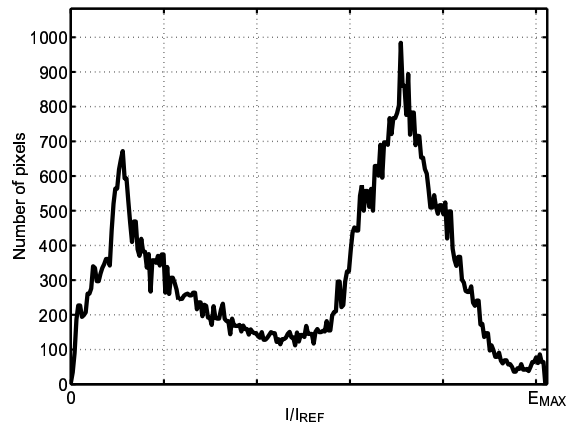
(b) Histogram.

Figure 3-19: Sample image used to illustrate the effects of different integra-
tion slot sets on the image sensor performance.

used. The following parameters are affected by the choice of integration slot set:

- Per–pixel memory requirements increase with the number of pixel checks
  (Equation 3.8). For the example $B_1 = 4$ bits, $B_2 = 3$ bits, $B_3 = 2$ bits and $B_4 = 1$ bit.
  Though apparently modest, a per–pixel memory increase of one bit in a mega–pixel
  sensor implies the addition of about 128KB of possibly on–chip memory.

  A potential advantage of a wider memory is that it can provide an adequate loga-
  rithmic representation of the scene being imaged before the integration cycle ends
  (the memory bus is free most of the time since the sensor requirements are only a
  finite number of bursts). Figure 3-19[7] shows a sample image that was processed with

---
[7]Image courtesy of Nicole S. Love.

the different integration slot sets. Figure 3-20 shows the resulting memory contents (viewed as indexes to a gray–scale color map). Clearly more integration slots offer a more detailed representation that can be used to control a mechanical iris, provide pre–scaling information for power–aware image processing algorithms, provide early data for crash avoidance and detection, etc.

- The intensity bins increase by R around the transition regions of the transfer characteristic so if a scene has details of interest around these areas, artifacts and severe image quality degradation can occur when $R \gg 0$ as the quantization noise increases significantly in an abrupt manner. Figure 3-21 shows the result of edge detection on images processed with $\mathcal{E}_4$ (top) and $\mathcal{E}_1$ (bottom). The increased intensity quantization noise leads to "false" edges and consequently a more difficult shape extraction.

- Integration slot sets with fewer elements result in a lower signal–to–noise ratio for a wider intensity range so the resulting images are noisier.

- Equation 3.24 shows that integration slot sets of different size are interchangeable from the I2D quantization noise standpoint only when:

$$
p_{I_{REF}}(I) \begin{cases} = 0 & \frac{I}{I_{REF}} \in [1, 128], \\ \neq 0 & \text{otherwise.} \end{cases}
\tag{3.39}
$$

that is, when the normalized illumination is concentrated in the $[0, 1]$ and $[128, 256]$ ranges. For all scenes with other illumination statistics the average quantization noise decreases as more slots are used. In general, the I2D quantization noise is a non–increasing function of the integration slot set size: consider the interval $I/I_{REF} \in [E_k, E_{k+1}]$ of an integration slot set $\mathcal{T}_1$, $|\mathcal{T}_1| = M$ which is broken in two, $I \in [E_k, E_m]$ and $I \in [E_m, E_{k+1}]$ with $E_k < E_m < E_{k+1}$ to form $\mathcal{T}_2$, $|\mathcal{T}_2| = M + 1$.

(a) Integration slot set $\mathcal{E}_1$ used.
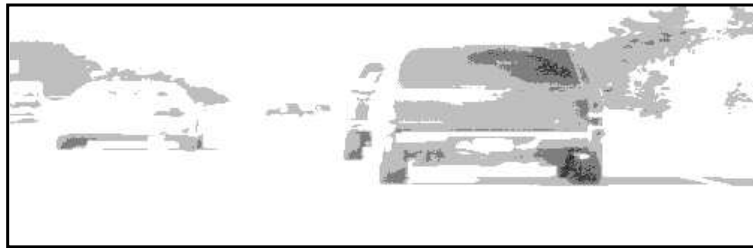
(b) Integration slot set $\mathcal{E}_2$ used.
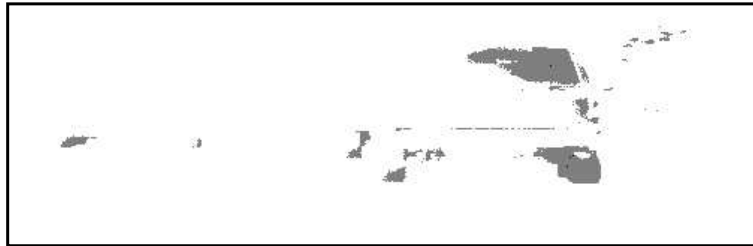
(c) Integration slot set $\mathcal{E}_3$ used.
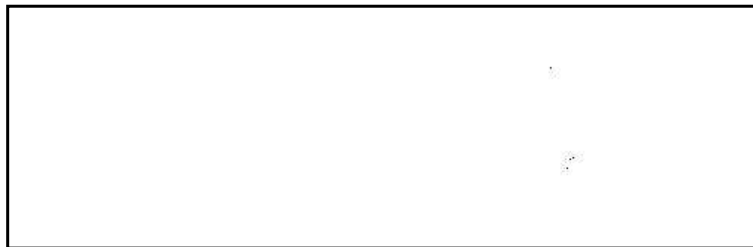
(d) Integration slot set $\mathcal{E}_4$ used.

Figure 3-20: Memory contents of processed sample image, viewed as indexes to a gray–scale color map. Pixels that use the longest integration slot are mapped to black, piixels that use shorter integration slots are mapped to brighter colors.

(a) Integration slot set $\mathcal{E}_4$ used.

(b) Integration slot set $\mathcal{E}_1$ used.

Figure 3-21: Edge detection results of sample image processed using two
different integration slot sets.

Then, using Equation 3.29 and denoting $\Delta E\left[\Delta I^2\right]_{UB}' = E\left[\Delta I^2\right]_{UB_{T_1}}' - E\left[\Delta I^2\right]_{UB_{T_2}}'$:

$$\Delta E\left[\Delta I^2\right] \quad \approx \quad E_{k+1}^2 \cdot \int_{E_k}^{E_{k+1}} p_{I_{REF}}\left(I\right) \cdot dI - \tag{3.40}$$

$$-\left(E_m^2 \cdot \int_{E_k}^{E_m} p_{I_{REF}}\left(I\right) \cdot dI - E_{k+1}^2 \cdot \int_{E_m}^{E_{k+1}} p_{I_{REF}}\left(I\right) \cdot dI\right) \tag{3.41}$$

$$\approx \quad E_{k+1}^2 \cdot \int_{E_k}^{E_{k+1}} p_{I_{REF}}\left(I\right) \cdot dI - \tag{3.42}$$

$$-E_m^2 \cdot \int_{E_k}^{E_m} p_{I_{REF}}\left(I\right) \cdot dI + E_{k+1}^2 \cdot \int_{E_m}^{E_{k+1}} p_{I_{REF}}\left(I\right) \cdot dI + \tag{3.43}$$

$$+E_{k+1}^2 \cdot \int_{E_k}^{E_m} p_{I_{REF}}\left(I\right) \cdot dI - E_{k+1}^2 \cdot \int_{E_k}^{E_m} p_{I_{REF}}\left(I\right) \cdot dI \tag{3.44}$$

$$\approx \quad \left(E_{k+1}^2 - E_m^2\right) \cdot \int_{E_k}^{E_m} p_{I_{REF}}\left(I\right) \cdot dI \geq 0 \tag{3.45}$$

Consequently if $|T_1| < |T_2| \Longrightarrow E\left[\Delta I^2\right]_{T_1} \geq E\left[\Delta I^2\right]_{T_2}$.
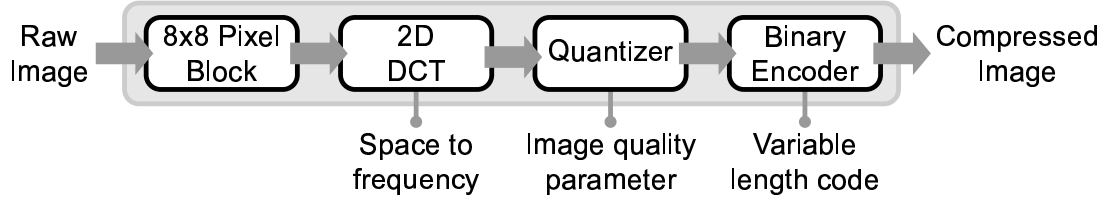
Figure 3-22: Block diagram of a JPEG image data compression chain.

3.8   Effects on Image Processing Tasks

The non–uniform bin size in the sensor transfer characteristic (Equation 3.12 and Figure 3-7) can have an impact on tasks performed by digital processors. Image data compression, a very important and often used task, was taken as a test case to explore the implications of using the novel wide dynamic range algorithm in a machine vision system.

One of the most popular image data compression methods was defined by the Joint Photographic Expert Group (JPEG) [61, 62]. The block diagram of the method can be seen in Figure 3-22. An image is first sectioned in 8×8 regions, to which a two–dimensional discrete cosine transform (DCT) is applied. The coefficients of the DCT are quantized by dividing them using the elements of a quantization matrix (Equation 3.46 shows an example of such a matrix). The resulting quantized coefficients are then encoded with a variable length code to obtain the compressed image.

$$Q_{50} = \begin{bmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\ 14 & 17 & 22 & 29 & 51 & 87 & 80 & 62 \\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{bmatrix} \tag{3.46}$$

Data compression is achieved because for typical natural scenes the magnitude of the DCT coefficients decreases rapidly in frequency. In addition, the human eye has lower sensitivity at high spacial frequencies, so the high frequency DCT coefficients (lower right corner in the quantization matrix) can be quantized more coarsely than the low frequency
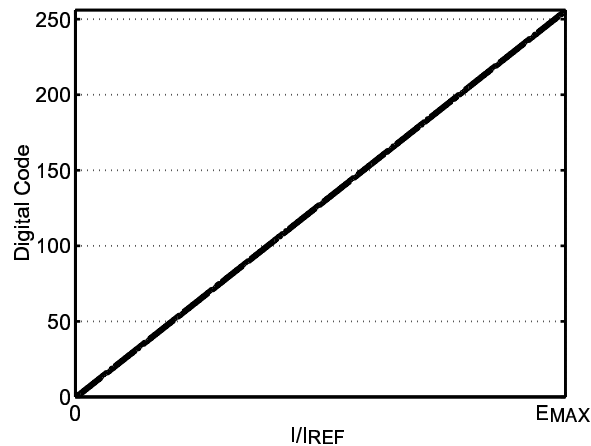
Figure 3-23: Ideal quantizer used as reference for the JPEG image data compression analysis. $N = 8$ bits.

coefficients (upper left corner in the quantization matrix) with little loss of perceived image quality. Consequently, after quantization many high frequency coefficients become zero and are therefore easier to code, while the low frequency coefficients undergo only minor changes.

The large bin sizes in the high illumination region of the predictive multiple sampling algorithm transfer characteristic may create false edges and therefore increase high frequency content of captured images. Data compression may be significantly affected by this added high frequency content, so to evaluate the magnitude of this effect the drafting office image was linearly quantized to 8 bits to create a baseline reference[8] (Figure 3-23). Then the multiple sampling algorithm was used to process the same data using increasing ADC resolutions, from $N = 4$ to $N = 7$. To keep the total pixel output fixed also at 8 bits the exposure ratio sets (and consequently the maximum pixel illumination I_{REF}) were adjusted accordingly as shown in Table 3.1. The resulting sensor transfer characteristics (Figure 3-24) show that coarser quantization occurs when more bits are produced by the multiple sampling algorithm and less bits are produced by the ADC.

The JPEG–compressed images using the different algorithm quantizers can be seen in Figures 3-25–3-28 for different quality factors[9]. Images showing the pixel–by–pixel difference between the reference image and the images processed with the multiple algorithm can

---

[8]This is an ideal, theoretical quantizer. If this quantizer were implementable there would be no need to have an illumination dynamic range expansion algorithm.

[9]The quality factor is a scalar that multiplies the quantization matrix. A higher quality factor translates into finer DCT coefficient quantization at all frequencies.

Table 3.1

Integration slot sets used in the JPEG data compression comparison

Set	N	$\mathcal{E}$	I_{MAX}/I_{REF}
1	4	$\{1, 2, 4, 8, 16\}$	$1/16$
2	5	$\{1, 2, 4, 8\}$	$1/8$
3	6	$\{1, 2, 4\}$	$1/4$
4	7	$\{1, 2\}$	$1/2$
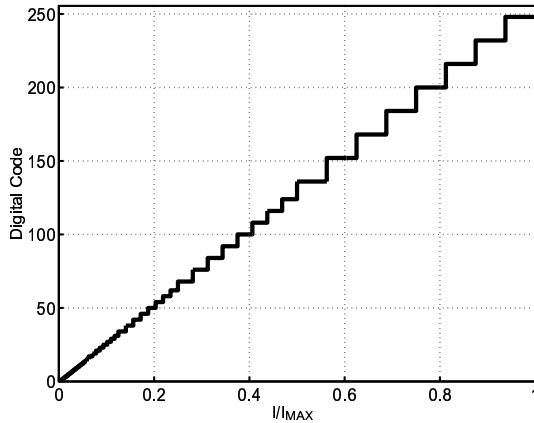
(a) $\mathcal{E} = \{1, 2, 4, 8, 16\}$, $N = 4$.
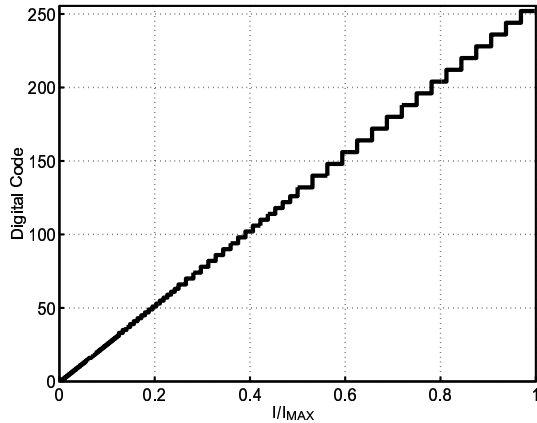
(b) $\mathcal{E} = \{1, 2, 4, 8\}$, $N = 5$.
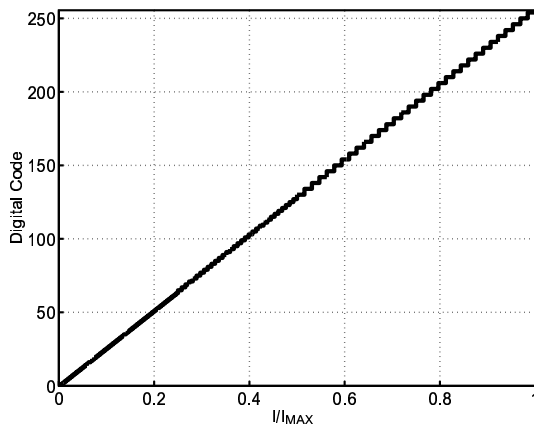
(c) $\mathcal{E} = \{1, 2, 4\}$, $N = 6$.
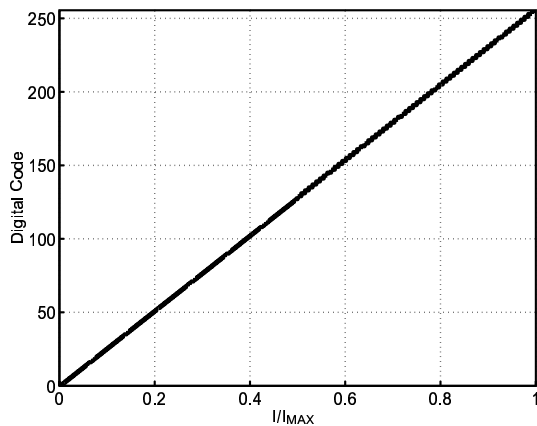
(d) $\mathcal{E} = \{1, 2\}$, $N = 7$.

Figure 3-24: Sensor transfer characteristics used in the image data compression analysis.

be seen in Figures 3-29–3-32. The differences in high illumination regions depend on the number of integration slots used: with less ADC resolution and more integration slots in-

(a) Quality factor 100.

(b) Quality factor 25.

Figure 3-25: JPEG–compressed images processed with $\mathcal{E} = \{1, 2, 4, 8, 16\}$ and $N = 4$.

(a) Quality factor 100.

(b) Quality factor 25.

Figure 3-26: JPEG–compressed images processed with $\mathcal{E} = \{1, 2, 4, 8\}$ and $N = 5$.

(a) Quality factor 100.

(b) Quality factor 25.

Figure 3-27: JPEG–compressed images processed with $\mathcal{E} = \{1, 2, 4\}$ and $N = 6$.

(a) Quality factor 100.

(b) Quality factor 25.

Figure 3-28: JPEG–compressed images processed with $\mathcal{E} = \{1, 2\}$ and $N = 7$.
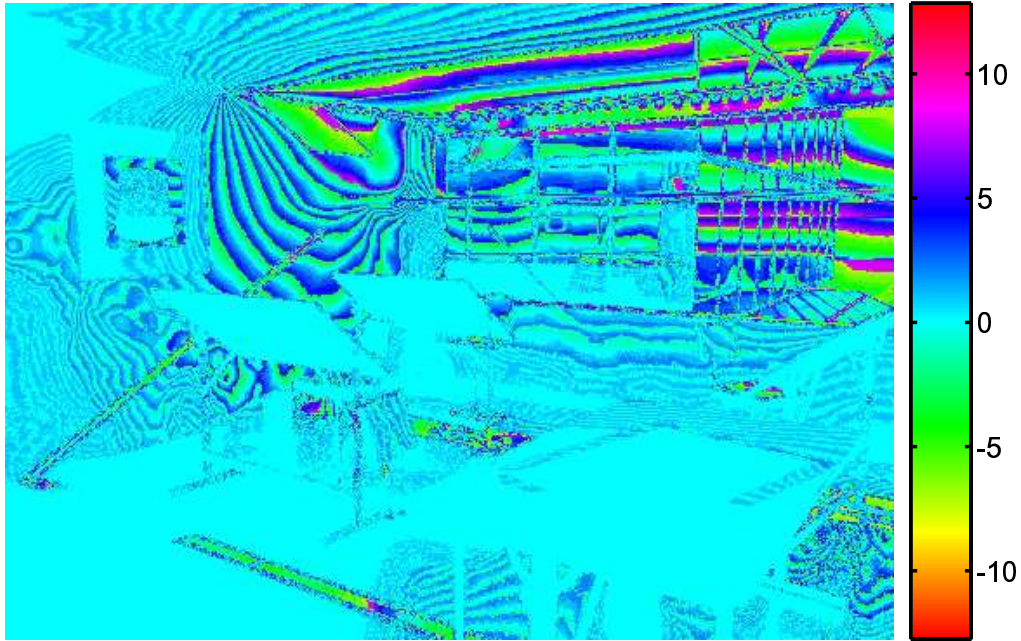
Figure 3-29: Differences between the reference image and the image captured with the multiple sampling algorithm having $\mathcal{E} = \{1, 2, 4, 8, 16\}$ and $N = 4$.
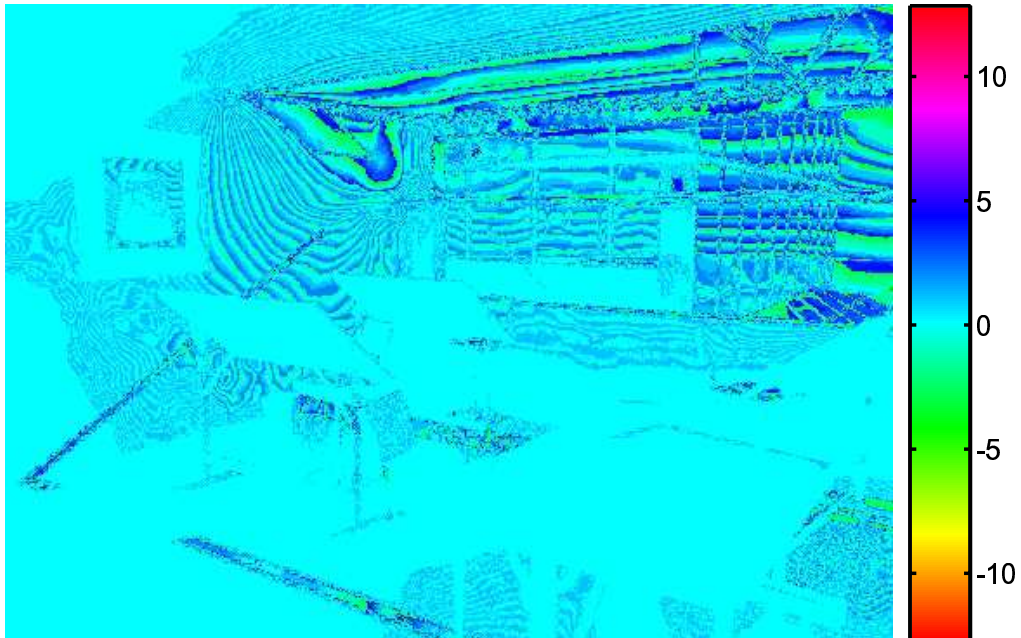
Figure 3-30: Differences between the reference image and the image captured with the multiple sampling algorithm having $\mathcal{E} = \{1, 2, 4, 8\}$ and $N = 5$.
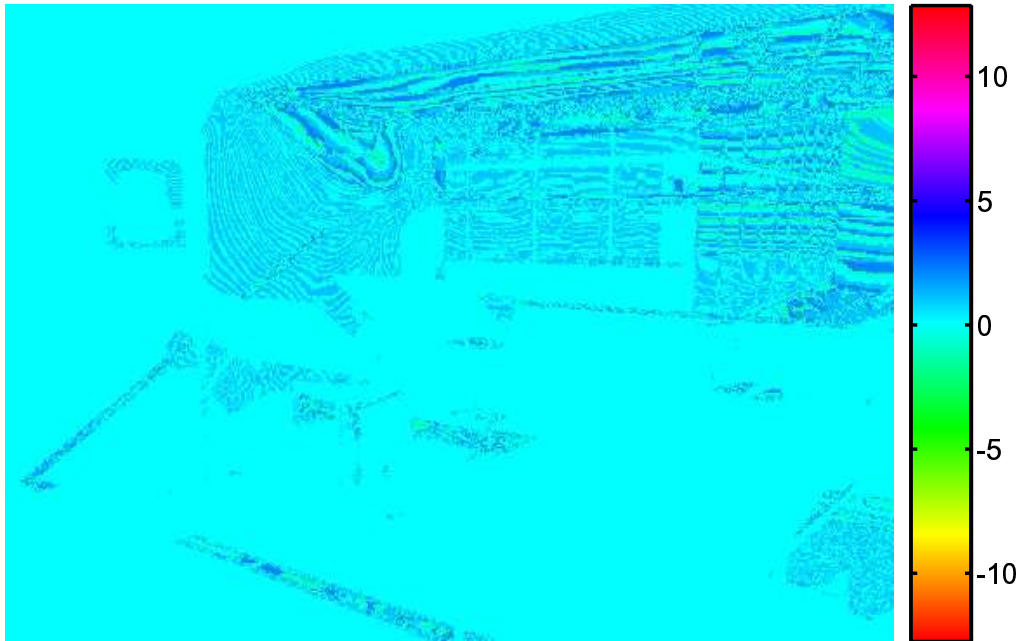
Figure 3-31: Differences between the reference image and the image captured with the multiple sampling algorithm having $\mathcal{E} = \{1, 2, 4\}$ and $N = 6$.
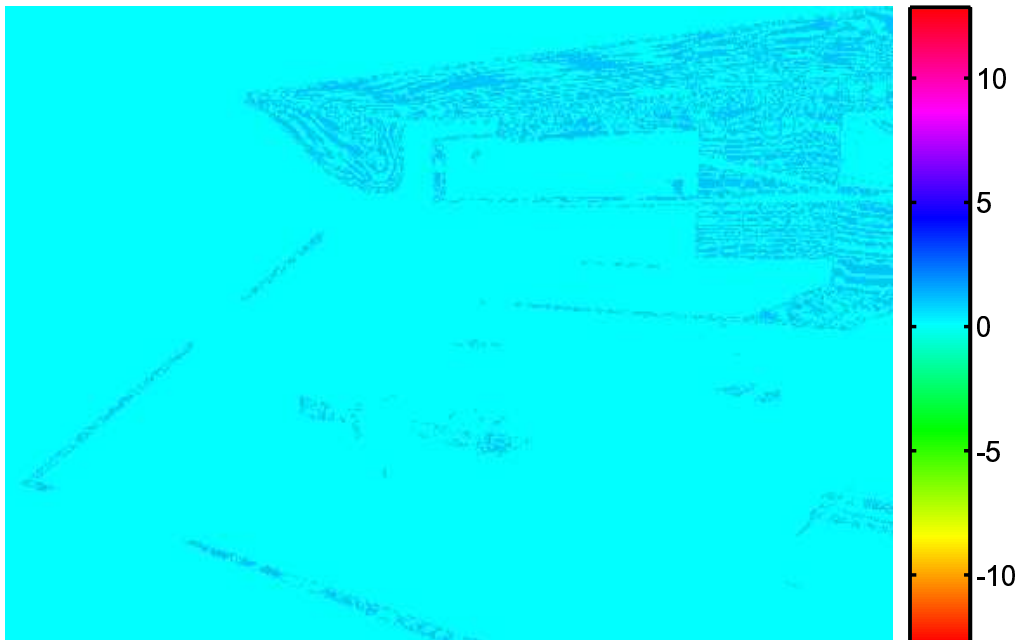
Figure 3-32: Differences between the reference image and the image captured with the multiple sampling algorithm having $\mathcal{E} = \{1, 2\}$ and $N = 7$.

Table 3.2
JPEG file size comparison for original draft office image

File	Quality							
	25		50		75		100	
	Size [B]	% Inc.	Size [B]	% Inc.	Size[B]	% Inc.	Size [B]	% Inc.
Ref.	3828	–	4935	–	6700	–	29474	–
1	3858	1	4970	1	6790	1	30424	3
3	3831	0	4942	0	6729	0	30009	1
2	3828	0	4941	0	6712	0	29629	1
4	3828	0	4935	0	6702	0	29485	0

creasingly wider high illumination regions are digitized using an effectively coarser quantizer. The difference between the coefficients of the two–dimensional DCT transform of the reference and algorithm–processed images can be seen in Figures 3-33–3-36, where as expected, images processed with more integration slots have more high frequency content (lower right corner) as compared to the reference. The JPEG encoder included in Matlab [60] was used to save both reference and algorithm–processed images for different quality factors: 100 (no data compression), 75, 50 and 25. Table 3.2 shows the file sizes and the percentage increase of the algorithm–processed images with respect to the reference. The data compression difference is reduced when:

- The ADC resolution increases and the number of integration slots is reduced to keep the total pixel output fixed at 8 bits. The transfer characteristic becomes that of a finer quantizer and thus differences at high frequency are minimized.

- The quality factor decreases. The coefficients of the DCT are themselves quantized, and the quality factor determines the quantization coarseness (higher quality factor, finer quantization), therefore for lower quality factors the quantized DCT coefficients of the reference and the algorithm–processed images tend to converge.

As Table 3.2 shows, for the draft office image the data compression difference is minimal ($< 3\%$) even for a quality factor of 100. However, these results depend on the scene statistics, scenes where the the majority of the probability is concentrated in the high illumination region will have a larger data compression difference. As further proof, the brightness–equalized draft office image (PDF in Figure 3-15 (f)) and both the original and brightness–equalized cubicle images (PDFs in Figure 3-15 (c) and Figure 3-15 (e)) were processed and the results are shown in Tables 3.3–3.5. Both brightness–equalized images
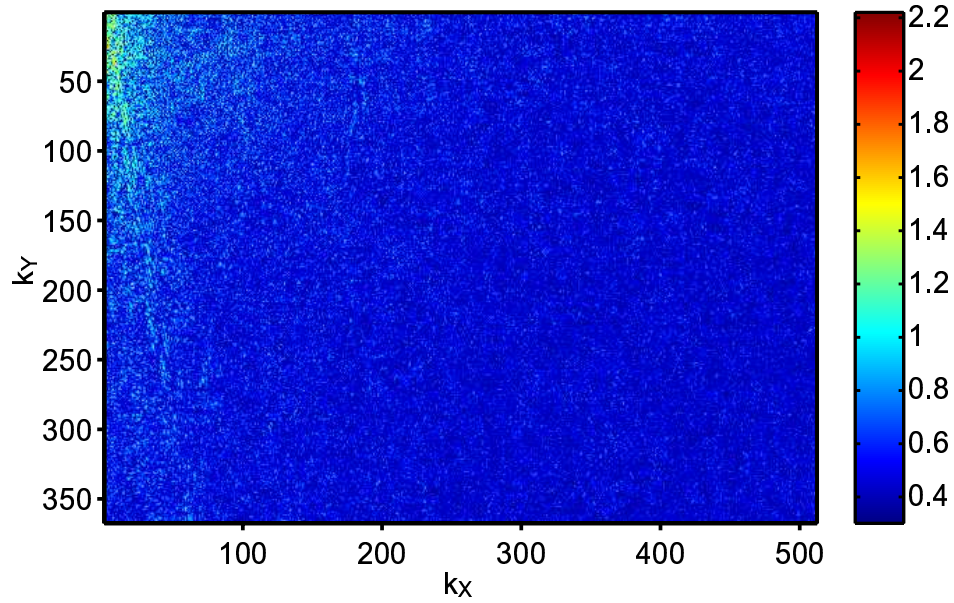
Figure 3-33: Difference of the two–dimensional DCT between the reference image and the image processed by the multiple sampling algorithm with $\mathcal{E} = \{1, 2, 4, 8, 16\}$ and $N = 4$. Magnitude of the differences in logarithmic scale.
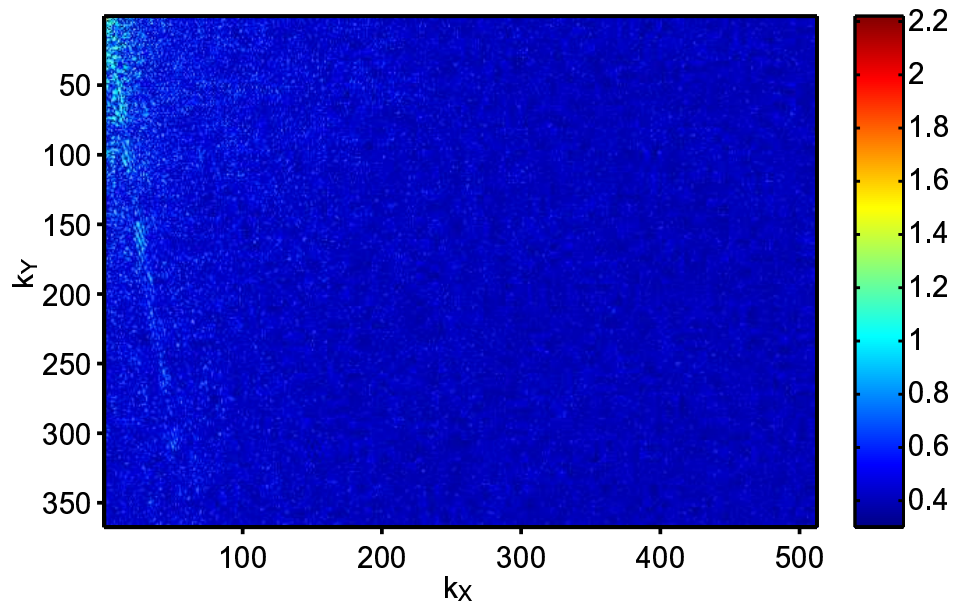
Figure 3-34: Difference of the two–dimensional DCT between the reference image and the image processed by the multiple sampling algorithm with $\mathcal{E} = \{1, 2, 4, 8\}$ and $N = 5$. Magnitude of the differences in logarithmic scale.
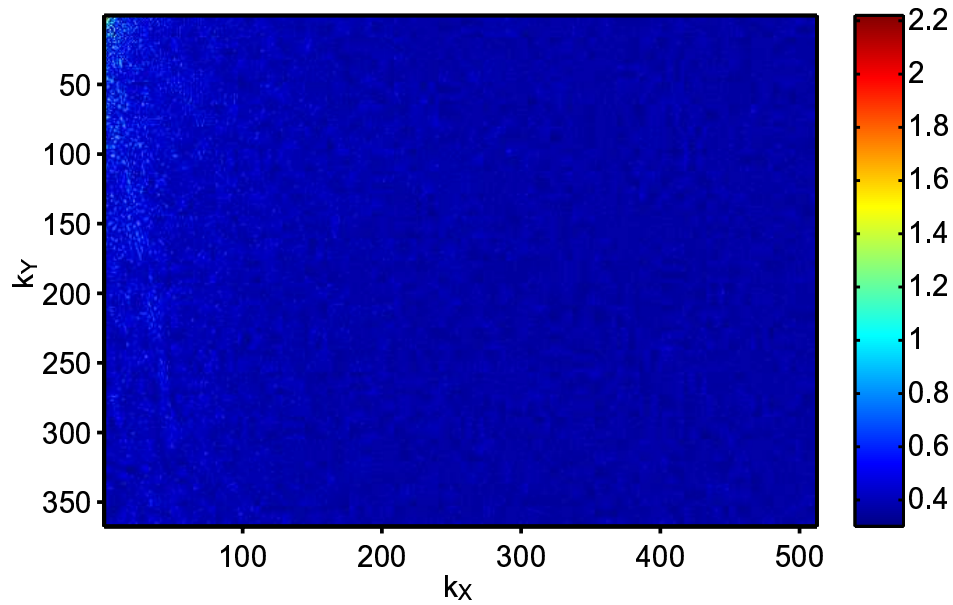
Figure 3-35: Difference of the two–dimensional DCT between the reference image and the image processed by the multiple sampling algorithm with $\mathcal{E} = \{1, 2, 4\}$ and $N = 6$. Magnitude of the differences in logarithmic scale.
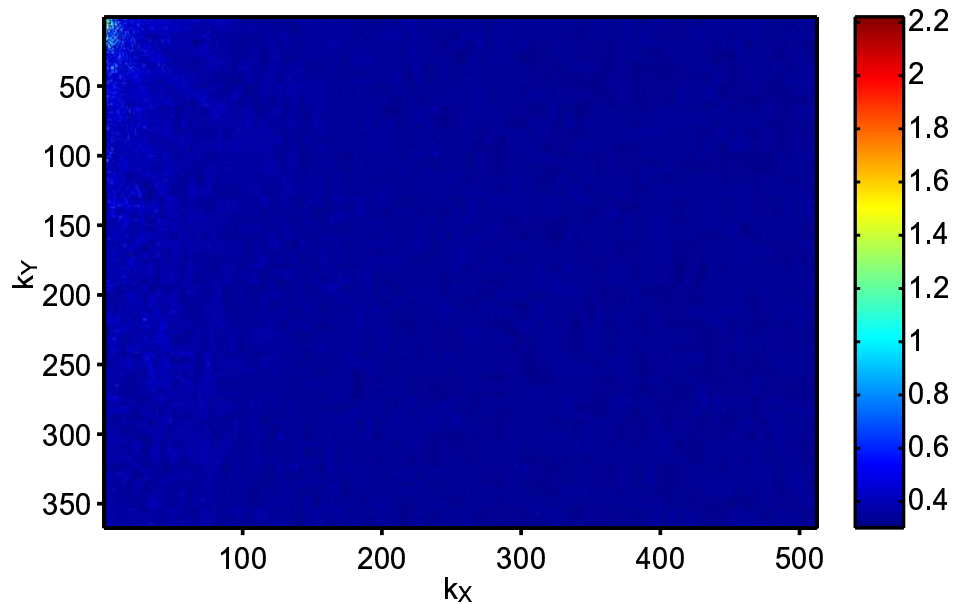
Figure 3-36: Difference of the two–dimensional DCT between the reference image and the image processed by the multiple sampling algorithm with $\mathcal{E} = \{1, 2\}$ and $N = 7$. Magnitude of the differences in logarithmic scale.

Table 3.3

JPEG file size comparison for brightness–equalized draft office image

File	Quality							
	25		50		75		100	
	Size [B]	% Inc.	Size [B]	% Inc.	Size[B]	% Inc.	Size [B]	% Inc.
Ref.	8864	–	12770	–	18107	–	77720	–
1	9073	2	13201	3	19165	6	86466	11
3	8960	1	12911	1	18519	2	83204	7
2	8885	0	12830	0	18222	1	80167	3
4	8864	0	12793	0	18137	0	78324	1

Table 3.4

JPEG file size comparison for original office cubicle image

File	Quality							
	25		50		75		100	
	Size [B]	% Inc.	Size [B]	% Inc.	Size[B]	% Inc.	Size [B]	% Inc.
Ref.	3027	–	3830	–	5304	–	25330	–
1	3028	0	3836	0	5319	0	25786	2
3	3030	0	3833	0	5310	0	25458	1
2	3027	0	3830	0	5304	0	25348	0
4	3027	0	3830	0	5307	0	25330	0

Table 3.5

JPEG file size comparison for brightness–equalized office cubicle image

File	Quality							
	25		50		75		100	
	Size [B]	% Inc.	Size [B]	% Inc.	Size[B]	% Inc.	Size [B]	% Inc.
Ref.	8831	–	13042	–	18648	–	71796	–
1	9004	2	13379	3	19214	3	78798	10
3	8904	1	13200	1	18959	2	76270	6
2	8860	0	13108	1	18829	1	73883	3
4	8841	0	13082	0	18671	0	72359	1

have more probability at high illumination, therefore they have a bigger data compression difference (around 10%) than the original images, whose probability is concentrated in the low illumination region. Of note is the fact that for a typical quality factor of 75 the data compression difference is only 6% or less for any of the selected images.

3.9   Summary

A novel predictive multiple sampling algorithm was introduced. The algorithm allows for integration periods (slots) of different duration to run concurrently by performing a predictive pixel saturation check at the potential start of every integration slot. The check relies on the assumption that the pixel illumination remains constant throughout the integration time. Other important characteristics of the algorithm are:

- The sensor requires pixels with non–destructive and conditional–reset capabilities, per–pixel storage and the implementation of an integration controller.

- The sensor transfer characteristic is linear and made of sections which have increasing illumination bins. To guarantee a monotonic sensor transfer characterisitic the integration slot ratios are effectively limited to powers of 2.

- The resulting signal–to–noise ratio has a distinctive "sawtooth" shape in the high illumination region but the low illumination region remains unaltered with respect to the SNR of a pixel without the dyanmic range expansion algorithm.

- The maximum dynamic range increase depends on the location, performance and implementation of both integration controller and memory.

- Given a certain integration slot set size, the optimal integration slot lengths which minimize the average illumination–to–digital code error depend on the illumination statistics. The precise composition of the optimal set can be obtained by exhaustively evaluating a computationally–friendly approximation of the I2D error.

- The optimal integration slot size mainly depends on the system resources available, tolerable SNR and desired maximum I2D error (for given illumination statistics).

- Data compression ratios are illumination–dependent but in general poorer than those of an ideal wide dynamic range image sensor with a uniform quantizer. However, for typical quality factors the difference is modest.

Chapter 4

Experimental Chip

4.1 Overview

A proof–of–concept integrated circuit was fabricated in a CMOS 0.18μm 1.8V 5–metal layer process with linear capacitor (double polysilicon) and 3.3V 0.35μm transistor options[1]. The 1.8V devices were used in all digital circuitry while the 3.3V devices were used in all analog circuitry. The IC die is 7600μm $\times$ 11700μm in size, its micrograph can be seen in Figure 4-1.

A block diagram of the image sensor is shown in Figure 4-2. The major components of the chip are a pixel array, a memory array, an integration controller vector and an analog–to–digital converter/correlated double sampled (ADC/CDS) vector. Supporting components include pixel and memory row decoders, memory and converter output digital multiplexers, pixel–to–ADC/CDS analog multiplexer and test structures.

Light intensity information is captured by pixels in the sensing array, and their output is routed through an analog multiplexer to the ADC/CDSs for quantization. The integration controller implements the dynamic range expansion algorithm, conditionally resetting pixels based on their output and their associated integration slot information which is stored in the memory.

---

[1]Fabrication process provided by National Semiconductor Corporation.
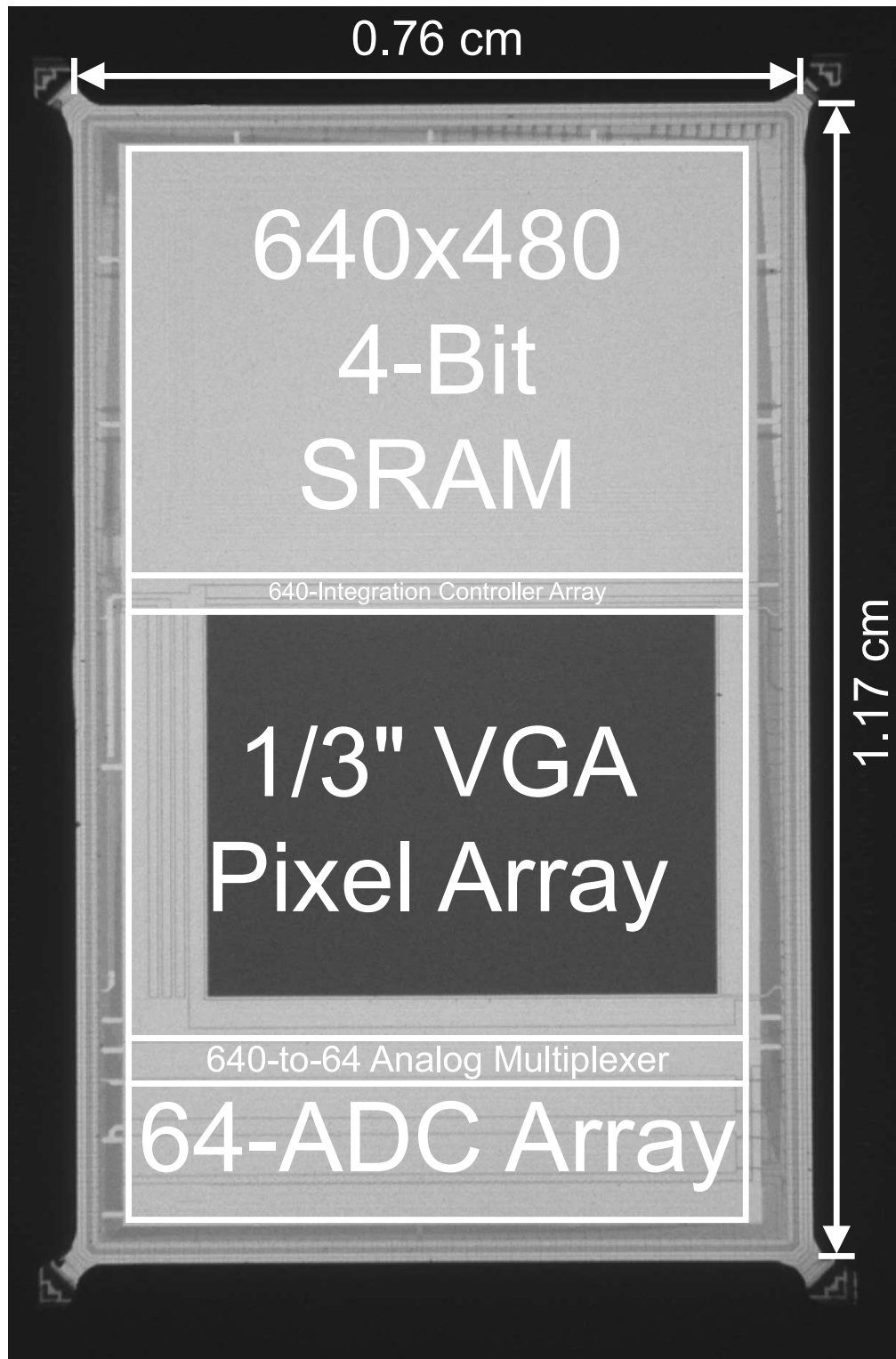
Figure 4-1: Proof–of–concept integrated circuit micrograph.
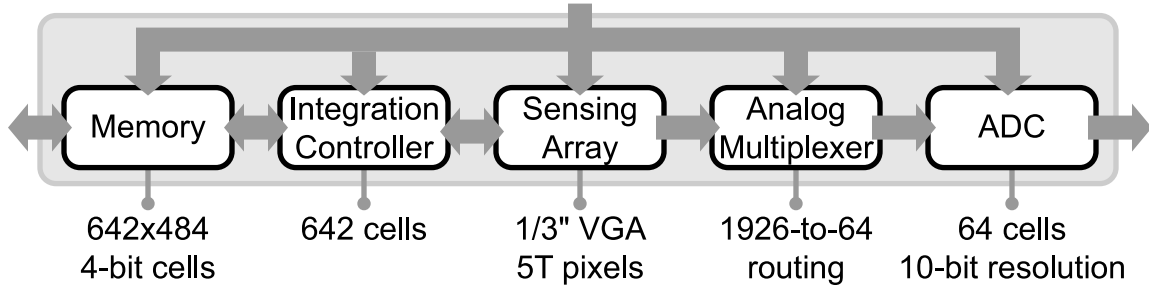
Figure 4-2: Proof–of–concept integrated circuit block diagram.
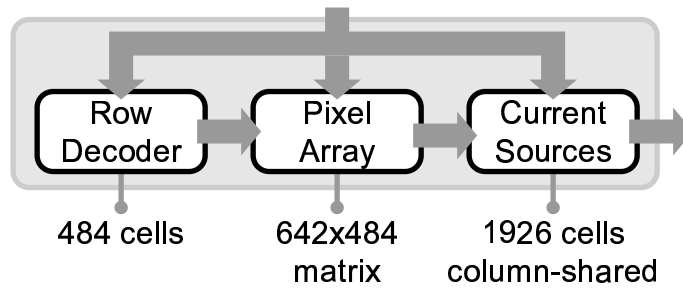
Figure 4-3: Sensing array block diagram.

4.2   Sensing Array

This block includes a pixel matrix, a row decoder and a vector of current sources (Figure 4-3). The sensing array was designed to be used with a $1/3''$ C(S)–Mount lens. This industry standard format calls for an array size of 4.8mm × 3.6mm with a diagonal of 6mm. The nominal spacial resolution chosen was VGA (640×480) thus pixels are squares of 7.5μm on the side. The actual resolution was slightly increased to 642×484 so as to tolerate small lens misalignments and allow for spacial stabilization of fabrication process parameters within the array. Fast read–out of a small region of interest (ROI) is provided as each pixel column has three output lines (Figure 4-4). A pixel at location (y, x), row $y \in [0, 483]$ and column $x \in [0, 641]$ is connected to output line $\mathtt{OUT}_k$, $k \in [0, 1925]$ following the relationship:

$$k = 3 \cdot x + (y \ mod \ 3) \tag{4.1}$$

where mod is the modulus function and $(y = 0, x = 0)$ is located at the upper left corner of pixel array.
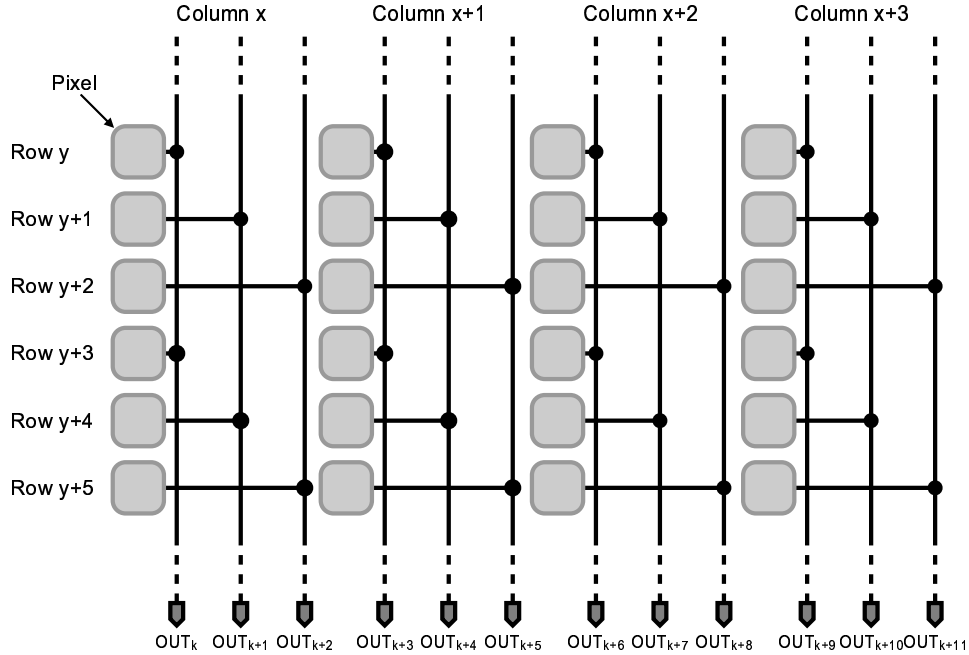
Figure 4-4: Pixels connections to output column lines.

4.2.1   Pixel

The pixel topology used in the proof–of–concept integrated circuit is a 5 NMOS transistor cell as shown in Figure 4-5. The photodiode is an n–diffusion/p–substrate type with a salicided exclusion mask. The top–most metal layer was used to route power (V_{EE}) and to shield light, covering the entire pixel area except for the photodiode. The pixel fill factors are 39% (exposed photodiode area in relation to total pixel area) or 49% (total photodiode area in relation to total pixel area). Transistors $M1$ and $M3$, both of which connect to the sensing node ("SNS" in Figure 4-5), are slightly longer than the other transistors in the pixel to decrease the subthreshold current when they are in the cut–off regime. A grounded substrate connection (p^+ plug) is also included to minimize optical crosstalk between neighboring pixels. The metal 4 layer routes all output (vertical) lines to minimize their parasitic capacitance, while the metal 3 layer routes all control (horizontal) lines.

Transistor $M1$ can be used as a charge spill gate to increase the sensitivity of the pixel. With the proper voltage in the **SHUTTER** control line, it acts as common gate amplifier pinning the photodiode voltage and allowing photo–generated subthreshold current flow from its source, a relatively high capacitance node (the photodiode), to a relatively small capacitance sensing node ("SNS" in Figure 4-5) [32]. $M1$ also provides electronic shutter
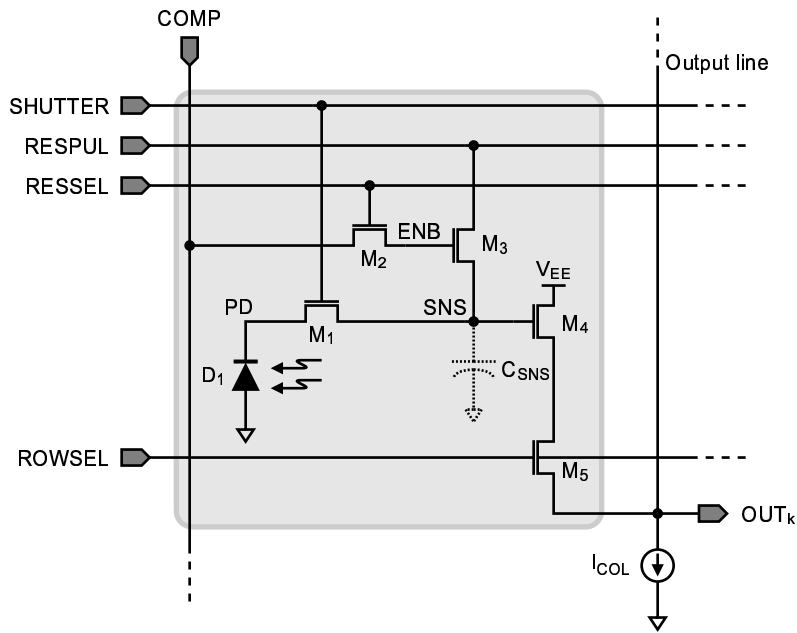
(a) Schematic. Current source shared by pixels in the column.
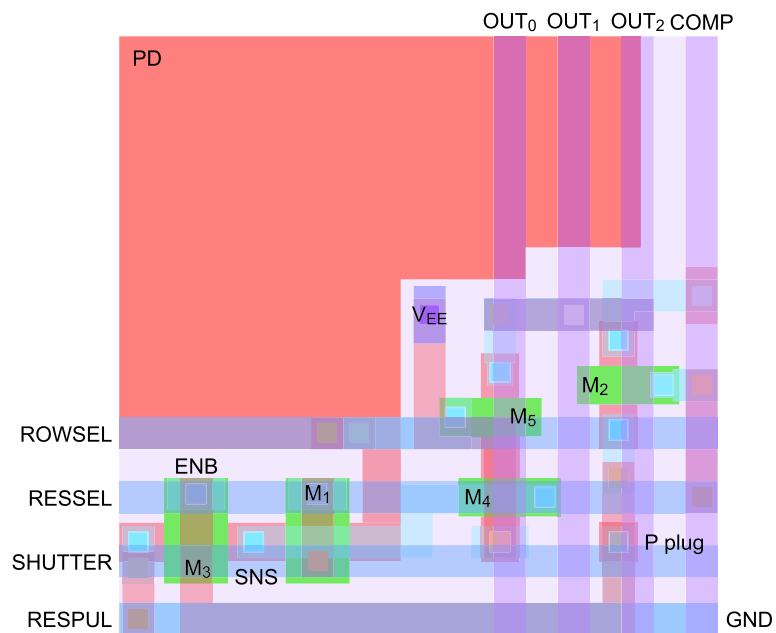
(b) Layout.

Figure 4-5: Pixel design used in the proof–of–concept integrated circuit.
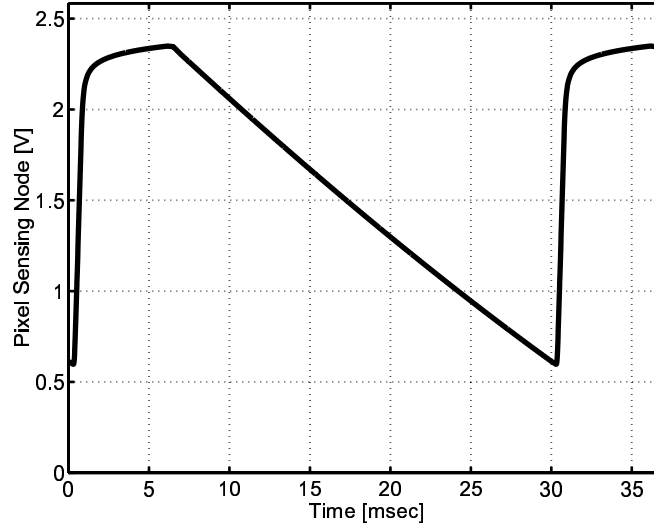
Figure 4-6: Typical evolution of the pixel sensing node voltage when the photodiode receives constant light intensity.

capabilities when the voltage at the `SHUTTER` control line drops significantly below the transistor threshold voltage. Transistors $M2$ and $M3$ together with control lines `RESSEL` and `COMP` provide the needed conditional reset capability [63]. Transistor $M4$ together with a shared column current source (I_{COL}) form a source follower amplifier that buffers the sensing node from the large capacitance of the column line. Transistor $M5$ is a switch that connects the source follower output to the column line when the pixel needs to be read.

The evolution of the sensing node voltage on a typical cycle for a pixel receiving constant light intensity can be seen in Figure 4-6. The cycle starts when the sensing node is reset during T_{RST} seconds, eliminating any visual information from the previous frame. The photodiode and sensing node are then isolated from the reset circuitry and allowed to collect (integrate) photo–generated charge (electrons) for T_{INT} seconds. At any point during this period of time the pixel can be accessed and have its signal non–destructively read. When the integration time ends, the shutter is closed and the sensing node is isolated from the photodiode thus its voltage remains constant and independent of further changes in light intensity. The pixel can then be read again to have its signal quantized by the ADC/CDSs.

Operating Modes

1. <u>Conditional Reset:</u> A pixel is selected for conditional reset when control signal `RESSEL` is HIGH. Transistor $M2$ is therefore on and there is an electrical connection between

the `COMP` control line and the gate of transistor $M3$ (node "ENB" in Figure 4-5) which is assumed to start the cycle at ground. The actual reset decision is encoded in the temporal evolution of `COMP`, if this line remains grounded throughout the cycle the pixel is not reset, while if this line goes HIGH, transistor $M3$ is turned on and there is a low impedance path between the photodiode node and the `RESPUL` line (Figure 4-7). `COMP` has to be LOW while `RESSEL` is still HIGH for a short period of time after the photodiode is reset to ensure that transistor $M3$ is turned off. Most reported designs use a constant reset voltage ("standard" reset), but if the drain of transistor $M3$ is tied to a control signal (`RESPUL`) a pulsed reset cycle can be used which diminishes photodiode soft reset problems [64]. In this scheme the photodiode is first grounded to erase any pixel memory and then charged up to a constant voltage as in the standard method. In this way the final photodiode reset level is constant regardless of the illumination received in the previous integration cycle. Since the HIGH voltages of the `RESSEL`, `RESPUL` and `COMP` control lines are set off–chip, the reset level of node V_{ENB} is:

$$V_{ENB_{RST}} = \begin{cases} V_{COMP_{HIGH}} & \text{if } V_{RESSEL_{HIGH}} \geq V_{COMP_{HIGH}} + V_T\left(V_{COMP_{HIGH}}\right), \\ V_S\left(V_{RESSEL_{HIGH}}\right) & \text{otherwise.} \end{cases}$$

(4.2)

where $V_S\left(V_G\right)$ is the maximum source voltage of a body–effected NMOS transistor when used to charge a high impedance node with gate potential V_G (Section A.1) and $V_T\left(V_{BS}\right)$ is the MOSFET threshold voltage for a bulk–to–source potential V_{BS} [65]. Consequently the reset level of the sensing node V_{SNS} is:

$$V_{SNS_{RST}} = \begin{cases} V_{RESPUL_{HIGH}} & \text{if } V_{ENB_{RST}} \geq V_{RESPUL_{HIGH}} + V_T\left(V_{RESPUL_{HIGH}}\right), \\ V_S\left(V_{ENB_{RST}}\right) & \text{otherwise.} \end{cases}$$

(4.3)

The off–chip voltages can therefore be raised above the analog power supply V_{EE} to partially or totally offset the reduction of the sensing node voltage swing introduced by the reset circuitry.

2. <u>Integration</u>: This mode is selected when the `RESSEL` line is LOW (node "ENB" in Figure 4-5 is assumed to be grounded). The voltage of the `SHUTTER` line ($V_{SHUTTER}$)
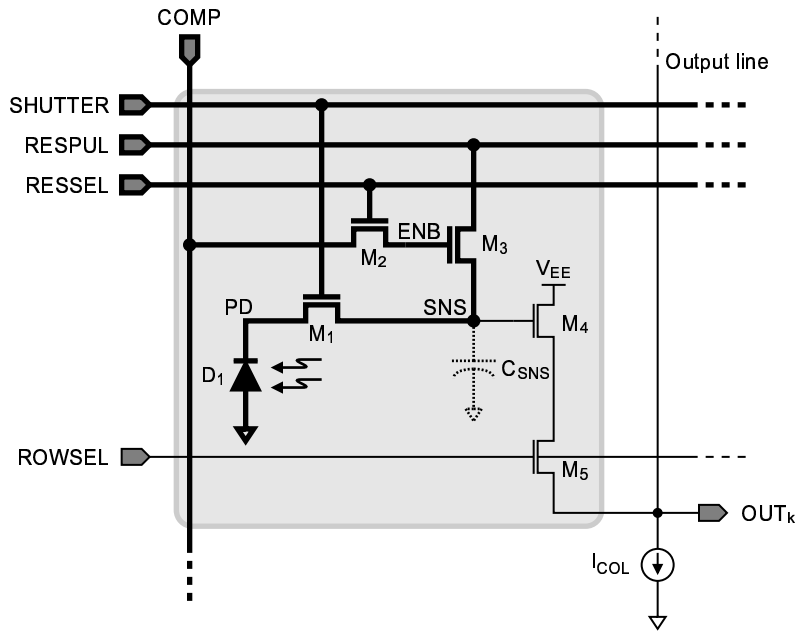
(a) Schematic.
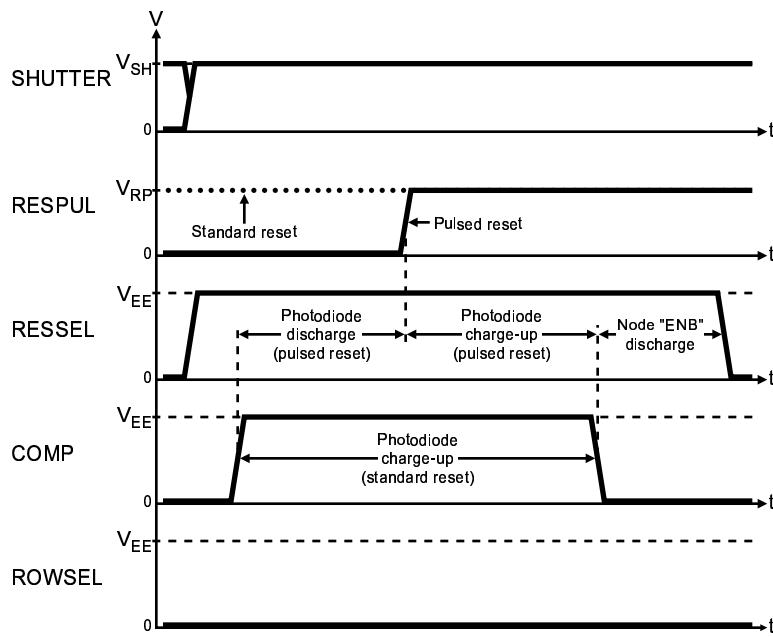
(b) Control phases.

Figure 4-7: Pixel in conditional reset mode.

is controlled off–chip and its actual level affects the pixel sensitivity, measured in volts per photo–generated electrons. If $V_{SHUTTER} \geq V_{SNS_{RST}} + V_T(V_{SNS_{RST}})$ then transistor $M1$ is always on and it is effectively eliminated from the pixel circuit as the photodiode and sensing node are connected by a low impedance path throughout the integration time. In this case, if C_{PD} is the capacitance associated with the photodiode and C_{SNS} is the capacitance associated with the sensing node "SNS", then the pixel sensitivity S_{PIXEL} is:

$$S_{PIXEL} \approx \frac{1}{C_{PD} + C_{SNS}} \approx \frac{1}{C_{PD}} \quad \text{as} \quad C_{PD} \gg C_{SNS} \tag{4.4}$$

On the other hand, if $V_{SHUTTER} \leq V_{SNS_{MIN}} + V_T(V_{SNS_{MIN}})$, where $V_{SNS_{MIN}}$ is the minimum sensing node voltage (mainly determined by the voltage offset of the pixel output source follower and the minimum ADC/CDS input voltage), then the photodiode is pinned at $V_{PD} \approx V_{SHUTTER} - V_T(V_{SHUTTER})$. Any photo–generated electron will produce a small decrease in V_{PD} which would induce a drain–to–source current to restore the photodiode voltage to its equilibrium value. This process draws charge (the same amount as was photo–generated) from node "SNS" so the pixel sensitivity in this case is:

$$S_{PIXEL} \approx \frac{1}{C_{SNS}} \gg \frac{1}{C_{PD}} \tag{4.5}$$

While the sensitivity is greatly increased by adding the cascode transistor, the charge transfer process across it is not instantaneous and therefore image lag is increased [66]. When the integration period ends, the SHUTTER control line can be grounded in order to cut off transistor $M1$ and isolate sensing node "SNS". The voltage there remains constant and can be subsequently read at a later time without alteration of the visual information represented by its magnitude.

3. <u>Read–out:</u> This mode is selected when the ROWSEL control line is HIGH. Switch transistor $M5$ is on, and transistor $M4$ and column current source I_{COL} form a common–drain amplifier that buffers node "SNS" into the column line OUT$_k$. After the settling time:

$$V_{OUT_k} = V_{OSF}(V_{SNS}) = V_{SNS} - V_{T_{N_0}} - \sqrt{\frac{2 \cdot I_{COL}}{\mu_n \cdot C_{OX} \cdot \frac{W}{L}}} - f(V_{SNS}) \tag{4.6}$$
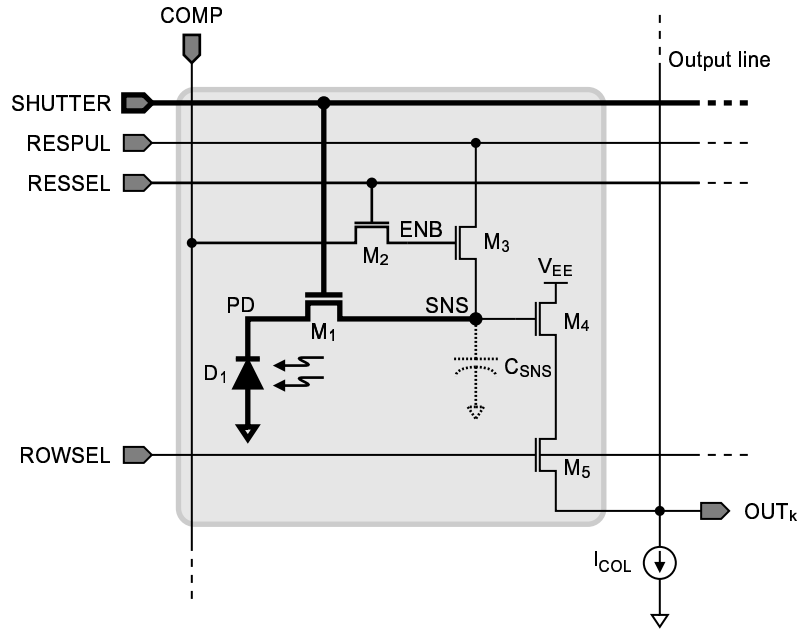
Figure 4-8: Pixel in integration mode.
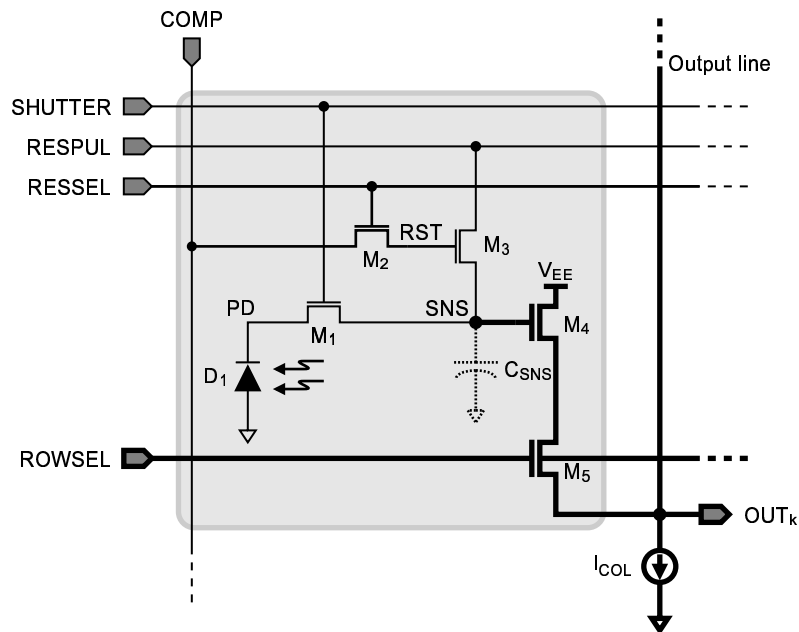
Figure 4-9: Pixel in read–out mode.

where $V_{O_{SF}}(V_G)$ is the output of a body–effected NMOS source follower with an input voltage V_G (Section A.2) and $f(\cdot)$ has a square root dependence on V_{SNS}, making the voltage shift non–linear.

The pixel can be read at any time, with the shutter open or closed. Due to the large
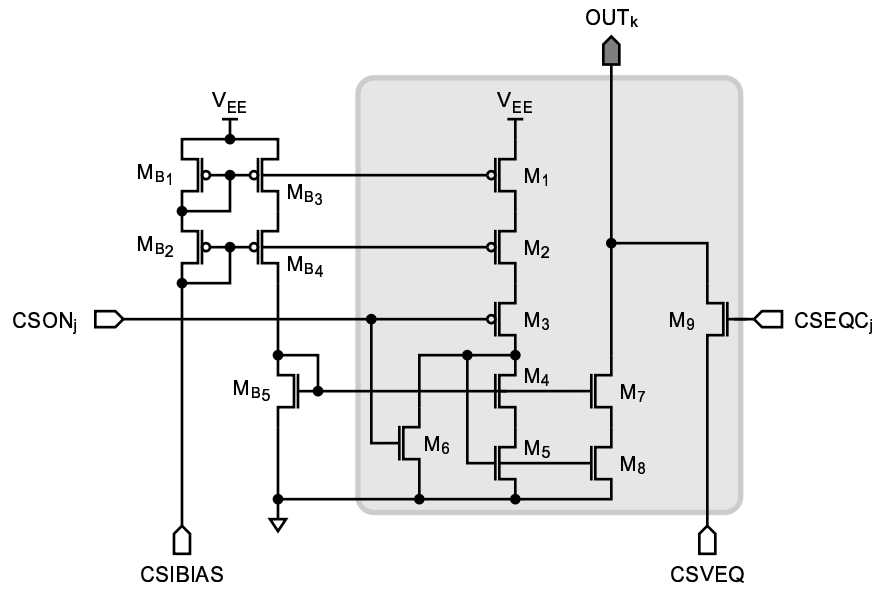
(a) Schematic (bias circuitry shared among all columns). $j \in \{0, 1, 2\}$.
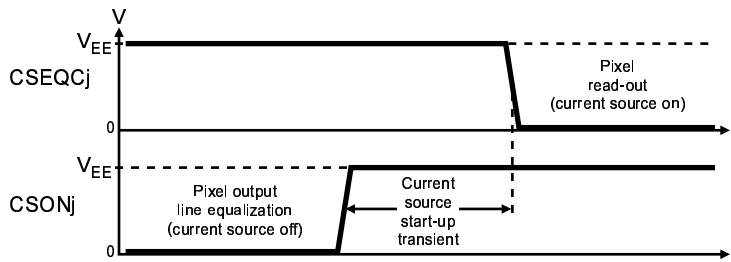
(b) Control phases.

Figure 4-10: Pixel array column current source.

difference in sampling time, it is typically read with the shutter open as part of the pixel saturation check of the wide dynamic range algorithm, and with the shutter closed for the pixel output quantization, at the end of the integration time.

4.2.2 Column Current Source

A high compliance cascoded current mirror was used to implement the pixel array column current sources as a reduced turn–on voltage was desired to maximize the allowable pixel voltage swing (Figure 4-10). Transistors $M1$–$M9$ are repeated at every pixel output

line (thus every pixel column has three current sources) while transistors M_{B_1}–M_{B_5}, which provide the required bias for the cascode, are shared among all column current sources. Signal CSIBIAS is the reference for the current mirror and needs to be provided off–chip, it has a nominal value of 1μA. The current mirror associated with output line OUT$_k$, $k \in [0, \cdots, 1925]$ can also be turned off with the CSONj signal, j=k mod 3, to minimize power dissipation when the pixels are not being read. Additionally, the output line can be equalized to the external voltage in the CSVEQ line when the control signal CSEQCj is HIGH.

The magnitude of the column current source affects both the settling time of the output line and the voltage offset of the source follower (Equation 4.6). This trade–off between speed and pixel voltage swing was broken with a scheme where the possibility of a pixel output line discharge is eliminated: before any pixel read–out its associated current source is off and its output line is charged to $V_{CSVEQ} \approx 250$mV, the lower bound of the allowable pixel output swing. The current source is then turned on and the pixel can be read, but under these conditions the transistor in the source follower always charges the output line to its final value. The current source, whose magnitude can now be drastically reduced, is left in the circuit to ensure that the source follower remains in its saturation regime. Figure 4-10(b) shows the phases of operation of the circuit. Overlapping the output line equalization turn–off and the current source turn–on before the pixel read–out can suppress any initial source transient and thus speed up the output line settling.

4.2.3   Row Decoder

The signals needed to command the pixels (SHUTTER, ROWSEL, RESSEL and RESPUL) have to be generated on a row–by–row basis with a circuit pitch–matched to the pixel height. An address pre–decoding scheme was adopted to reduce complexity at the row level and loading of the address lines [8,67]. The binary–represented pixel row address in the bus PROW<8:0> is used to generate eighteen pre–decoded internal address signals (PDRA<xxxxx>). Sixteen of them are arranged in four groups of four generated by 2–to–4 decoders while the remaining two pre–decoded signals form a fifth group which is generated by a buffer and inverter combination (Figure 4-11). Only one of the pre–decoded lines, which run through the rows of the pixel decoder core, is active at any particular point in every group. The address decoding process is completed by a 4–input AND gate shared by four pixel rows, and a 2–input AND gate at every pixel row (Figure 4-12). The 4–input AND gate takes input
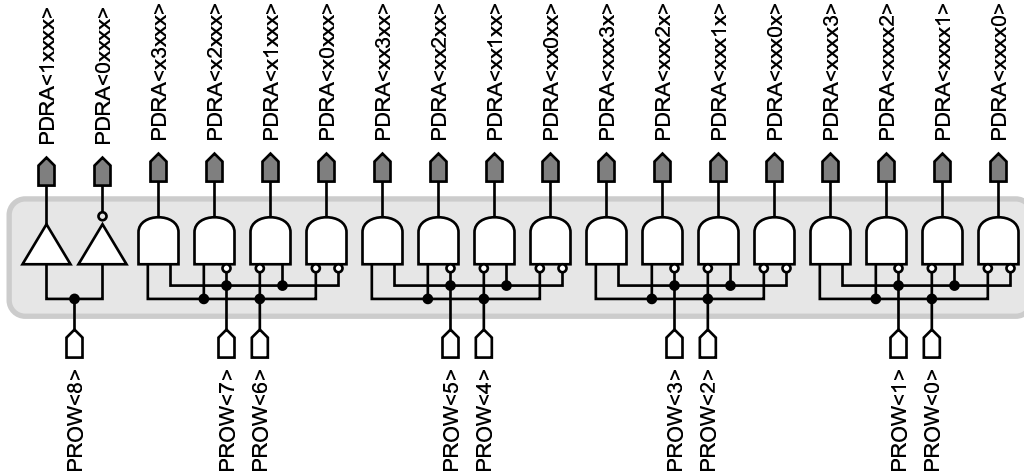
Figure 4-11: Sensing array row decoder address pre–decoder schematic.

from one line out of each of the five most significant groups (generated by `PROW<8:2>`) and thus selects four pixel rows. The final selection is made by the 2–input AND gate, whose inputs are the output of the 4–input AND gate and one line out of the least significant group (generated by `PROW<1:0>`). Consequently with this scheme each pre–decoded line only serves 121 AND inputs. Additionally, these two logic gates are small and suitable to be efficiently laid out in the small area defined by the pixel height.

The row address decoder core (shaded area in Figure 4-12), which is repeated at every pixel row, consists of decoding logic, control signal memory and buffering. The final address decoding result is stored at every row by a D–type flip–flop. The `PRDCLR` signal provides multiple row latching capabilities: when `PRDCLR` is LOW only the last row addressed is active, but all rows that are addressed while `PRDCLR` is HIGH become active and thus simultaneous control of them is possible. This functionality is achieved by disabling the clock of a particular register once the row associated with it is selected. Independent of the address bus contents, all rows can be made active with the `PRDON` signal, and inactive with the `PRDOFF` signal.

The gated row address decoding result ("ROW ENB" node in Figure 4-12) acts as an enable signal for the row memory elements associated with every pixel control signal. To minimize area these elements are set/reset latches with logic that makes it impossible to have the forbidden S=HIGH, R=HIGH combination. Their output can be changed when their row is selected and consequently `SHUTTER`, `ROWSEL`, `RESSEL` and `RESPUL` are set or reset
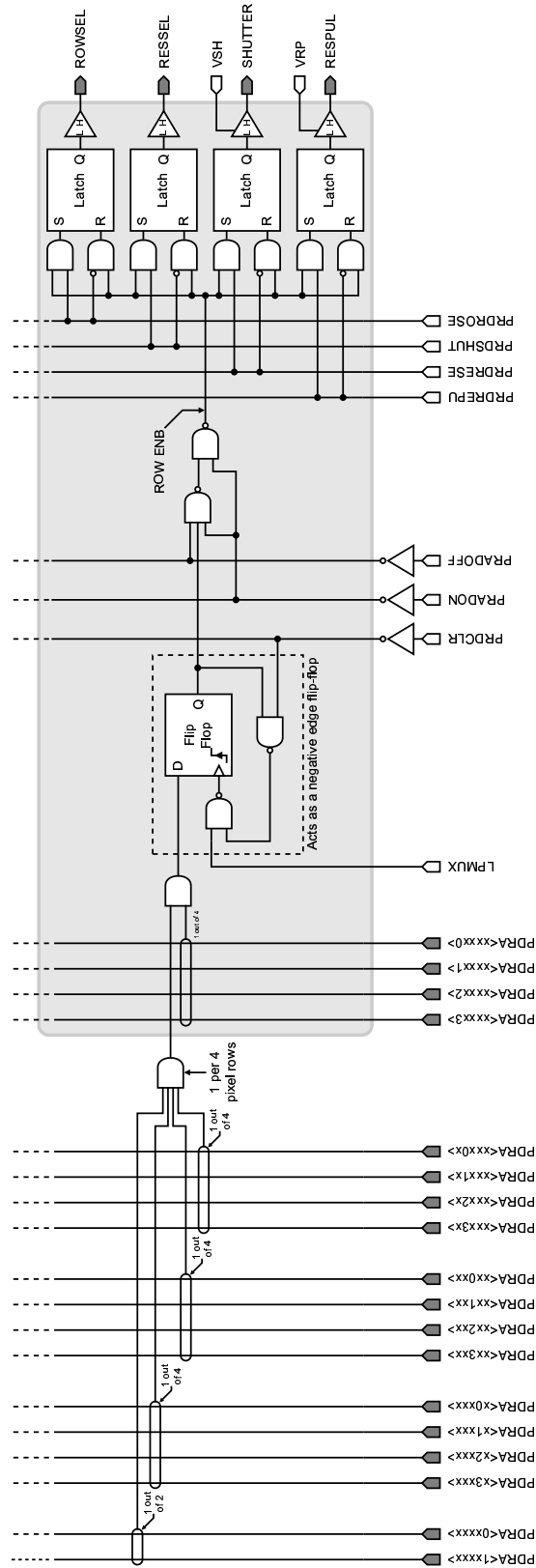
Figure 4-12: Sensing array row decoder schematic (row circuitry).

Table 4.1
Pixel row decoder global signals truth table

PRDON	PRDOFF	PRDCLR	Description
0	0	0	Single pixel row activation
0	0	1	Multiple row pixel activation
0	1	X	All pixel rows inactive
1	X	X	All pixel rows active

(a) Schematic.   (b) Symbol.
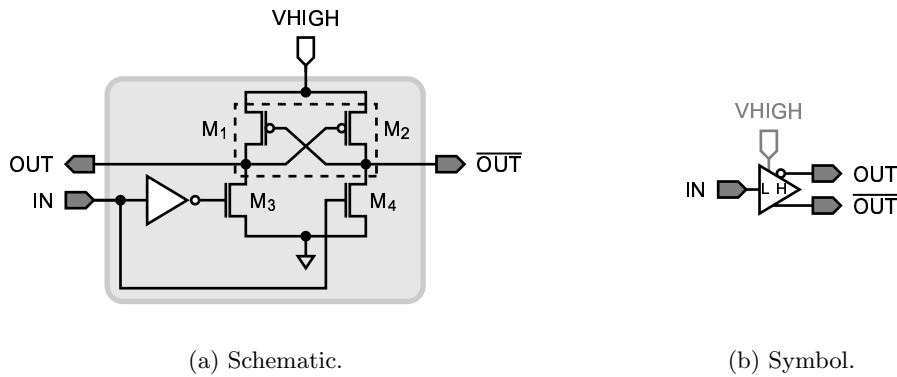
Figure 4-13: 1.8V to 3.3V digital level converter. Dashed box encloses 3.3V devices.

by the PRDSHUT, PRDROSE, PRDSEL and PRDREPU external signals, respectively. Figure 4-14 shows the behavior of the pixel row decoder under different control signals combinations.

To eliminate glitches, the decoder rows should be inactive (PRDON LOW and PRDOFF HIGH) during the row selection. PRDOFF should be deactivated only after this process is finished, and after the values of PRDSHUT, PRDROSE, PRDSEL and PRDREPU have settled to their final value.

A digital level converter is present at the output of the latches providing buffering and the interface between the digital (1.8V devices) and analog (3.3V devices) areas. The level converter is made of a cross–coupled 3.3V PMOS pair ($M1$ and $M2$), and two 1.8V NMOS pull–down devices ($M3$ and $M4$) which are sized so as to quickly overpower the cross–coupled pair (Figure 4-13). The circuit therefore has low short–circuit currents and fast state transitions without static power dissipation. The HIGH voltage for ROWSEL and RESSEL is set to the analog power supply (V_{EE}) while the HIGH voltage for SHUTTER and RESPUL is controlled off–chip by the VSH and VRP signals, respectively.
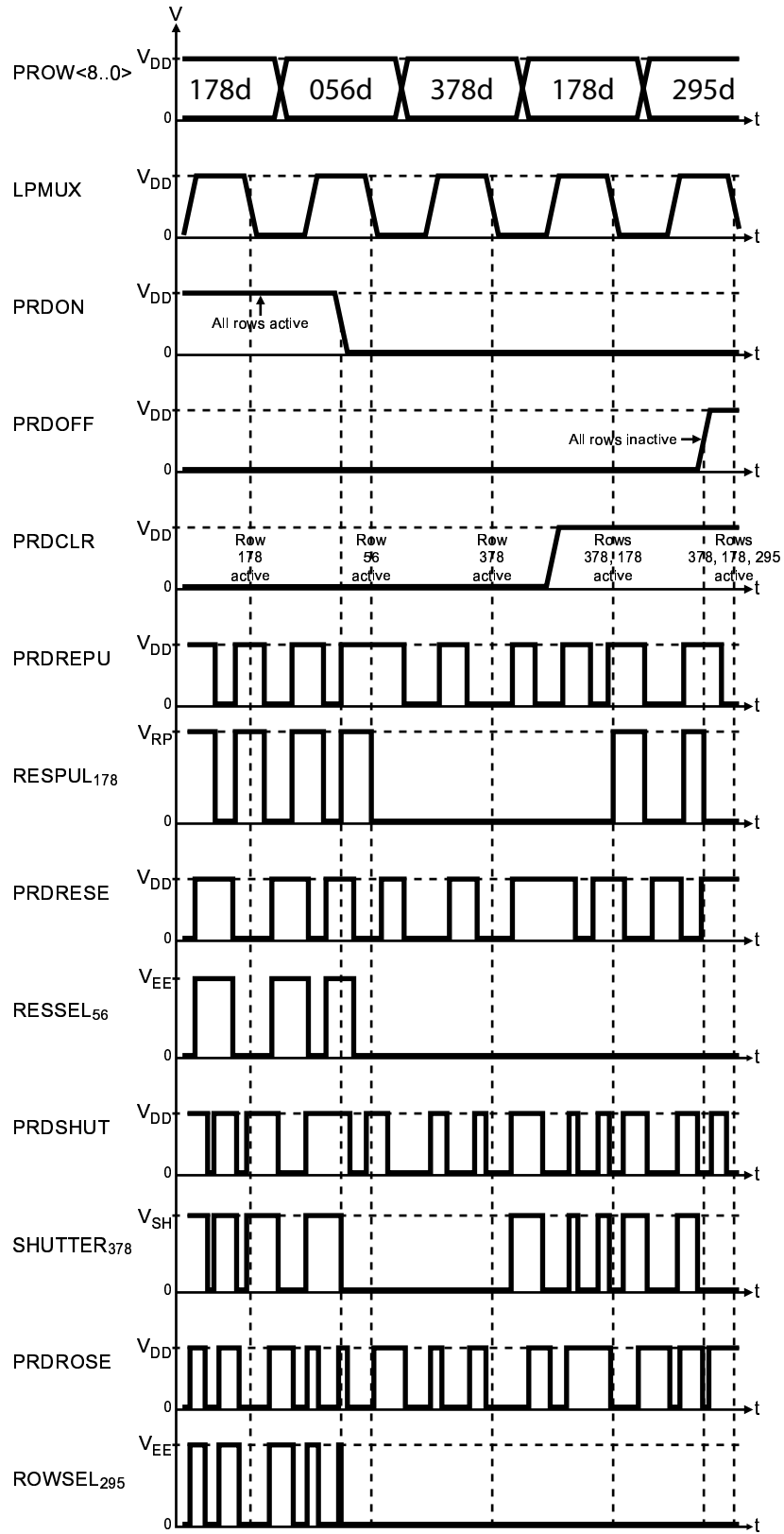
Figure 4-14: Pixel row decoder operating phases.

4.3 Analog Multiplexer

The 1926 pixel output lines are routed to the 64 ADC/CDS cells present in the IC by an analog multiplexer. A shift register–based approach was favored over a classical pass–gate chain design to minimize area, line parasitics and settling time of the pixel output lines (Figure 4-15). In this scheme the outputs of a 1920–stage shift register control a single pass gate at each pixel output line. The input of this register is the `AMDIN` signal while the output of the last stage is taken off–chip to the `AMQOUT` signal. The stages change output at the negative edge of `AMLATCH`. The multiplexer can be turned off with the `AMON` control signal, in this way the pixel output lines can be disconnected from the ADCs while the shift register is loaded to its final combination (thus avoiding glitches) or during the pixel check cycle (thus avoiding additional parasitic loading).

The first and last pixel columns are controlled independently of the core sensing array columns. The gating of the leftmost pixel column output lines is controlled individually by the `AMLED<2:0>` bus (Figure 4-15(a)) while the gating of the rightmost pixel column output lines is controlled individually by the `AMRED<2:0>` bus (Figure 4-15(c)).

The outputs of the core multiplexer AM_c, $c \in \{0, 1, \cdots, 1919\}$ are hard–wired to a single ADC input $a \in \{0, 1, \cdots, 63\}$, following the relationship $a = c \; mod \; 64$. From Equation 4.1, this means that the output of a pixel at location (y, x), row $y \in [0, 483]$ and column $x \in [1, 640]$ is digitized by converter $a(y, x)$ following the relationship:

$$a(y, x) = [3 \cdot x + (y \; mod \; 3)] \; mod \; 64 \tag{4.7}$$

The three output lines per pixel column allow for the simultaneous quantization of 64 pixels (with certain restrictions). With this scheme, for example, a 3×3 region can be digitized simultaneously, which is very advantageous for some image processing tasks. If $\mathcal{P}$ is defined as the set of all possible combinations of allowable 64 pixels to be digitized, then $p \in \mathcal{P}$ if any pair of pixels at locations $(y, x) \in p$, $(z, w) \in p$ satisfy:

1. $(y, x) \neq (z, w)$

2. $a(y, x) \neq a(z, w)$

The first column of the pixel array is connected to the first three (leftmost) ADCs/CDSs (Figure 4-15(a)) while the last pixel column is connected to the last three (rightmost)
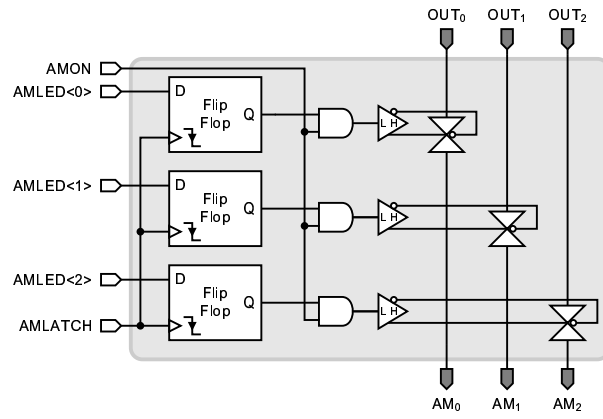
(a) Circuit in first pixel column.
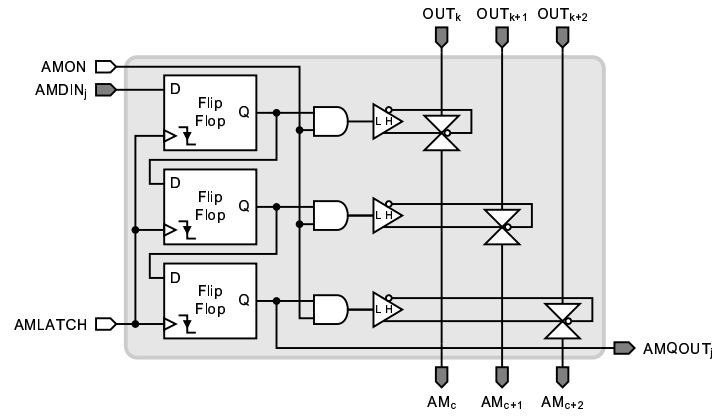
(b) Core circuit.
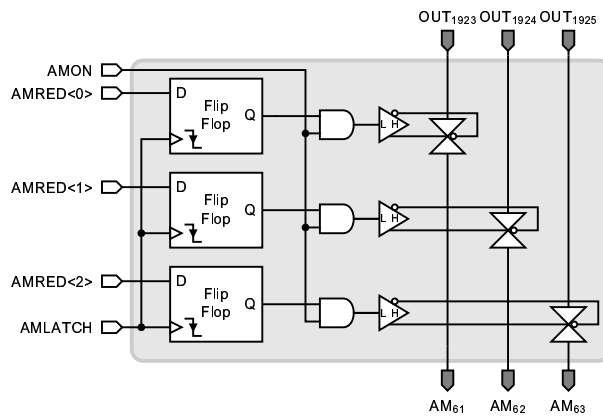
(c) Circuit in last pixel column.

Figure 4-15: Analog multiplexer schematic.
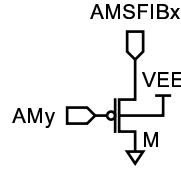
Figure 4-16: PMOS source follower used to provide analog readout of the ADC/CDS input voltages. $x \in \{0, 1\}$, $y \in \{0, 63\}$.

ADC/CDSs (Figure 4-15(c)). Off–chip readout of the input analog voltage of the ADC/CDSs was provided with two PMOS source followers (Figure 4-16), connected to AM_0 (output AMSFIB0) and AM_{63} (output AMSFIB1).

4.4 Integration Controller

Most of the circuitry needed to implement the conditional pixel check required by the novel wide dynamic range algorithm is implemented in this block. The integrated circuit has a column–parallel integration controller so that a complete row can be checked at a time thus reducing the frame check time. The circuitry present at each pixel column (shaded area in Figure 4-17) needs to be fast in order to further minimize the check time, and small, as it has to be pitched–matched to the pixel width. The column circuitry can be roughly divided in two: a circuit that determines whether or not the pixel output voltage is above or below an external reference voltage, and a circuit that determines whether or not the pixel was reset in the previous check cycle.

The output column corresponding to the pixel row that needs to be checked is expected binary–encoded in the CSEL<1:0> bus. From Equation 4.1 CSEL<1:0>= $k\ mod\ 3$ if row k needs to be checked as the column information is irrelevant. A 2–to–3 decoder shared by all the columns pre–decodes this information so that the final pixel output line decoding is made by transistors $M7$–$M9$. NMOS devices are used since the nominal maximum pixel output voltage (1.25V) is far below $V_{EE} - V_{T_N}$. The selected pixel output line is then compared to an external voltage reference CVREF by a standard dynamic voltage comparator. The first phase of operation, the sample phase, takes place when CSAMPLE is HIGH and CCOMP is LOW (Figure 4-18). In this configuration the reference voltage and the pixel output voltage are stored in the two internal nodes of the comparator. The second and last phase of operation, the comparison itself, takes place when CSAMPLE is LOW and CCOMP is HIGH.
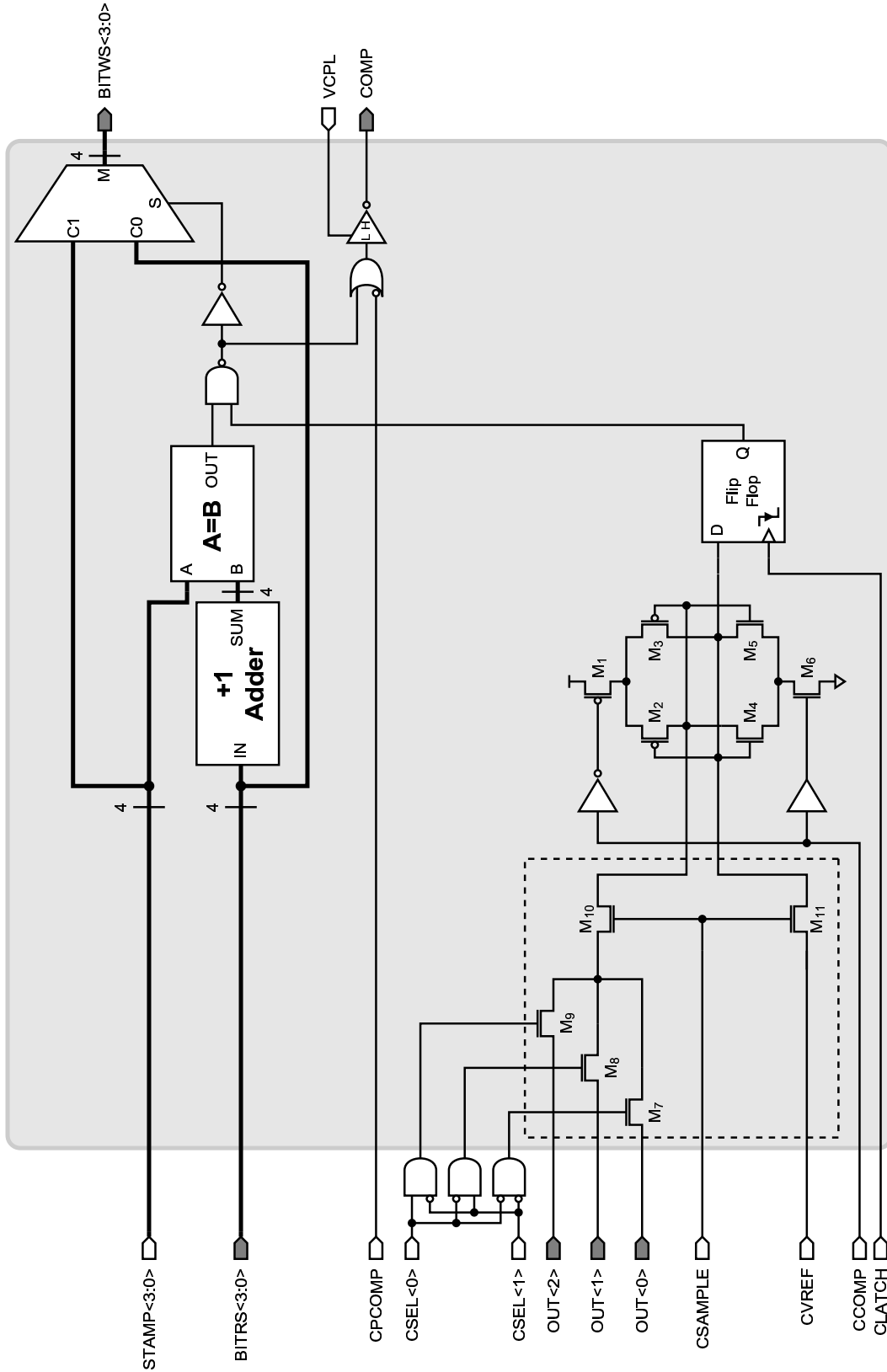
Figure 4-17: Integration controller schematic.
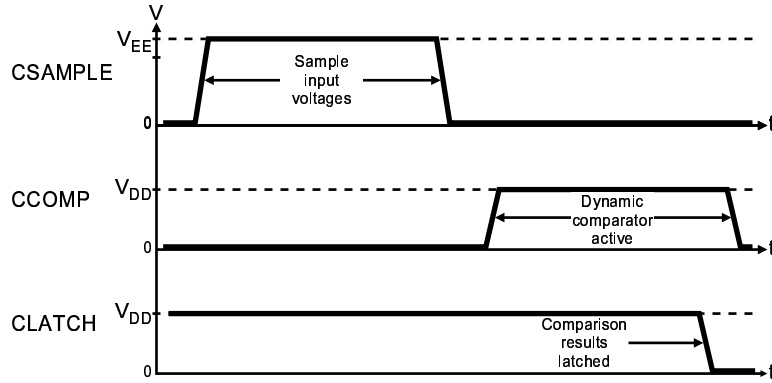
Figure 4-18: Integration controller dynamic comparator phases.

In this configuration a pair of cross–coupled CMOS inverters formed by transistors $M1$–$M6$ is activated, taking the internal nodes to the appropriate rails depending on their relative magnitude. The comparison result is stored in a D–type flip–flop to allow pipelined operation.

The circuit that determines whether the pixel was reset in the previous check cycle is made of a 4–bit +1 digital adder and a 4–bit digital comparator. The index associated with the pixel being checked, which is stored in an associated SRAM location, is expected in the BITRS<3:0> bus during a check cycle. One unit is added to this binary–encoded number by the digital adder, whose schematic is shown in Figure 4-19. Reduced area and fast operation were the two main objectives pursued with this particular implementation. The output does not overflow for the [0, 14] input range.

The output of the adder is compared to an externally generated time stamp, which should be the current check cycle number binary–encoded in TSTAMP<3:0>. The circuit used to generate this result is shown in Figure 4-20. A HIGH output indicates that the pixel was reset in the previous check cycle (BITRS=TSTAMP−1), while a LOW output indicates that the pixel was reset at least two check cycles in the past (BITRS<TSTAMP−1). The memory contents have to be zeroed before the acquisition of a new frame for proper operation during the first check cycle, so that when TSTAMP = 1 all pixels are potentially enabled for reset.

The pixel voltage has to be below the reference voltage and the last pixel reset cycle has to be the previous check cycle for a pixel to be reset. When this condition is true the column COMP line becomes a buffered version of the PCOMP external signal, otherwise it stays grounded. For maximum flexibility the HIGH voltage of the COMP signal is determined

(a) Schematic.   (b) Symbol.
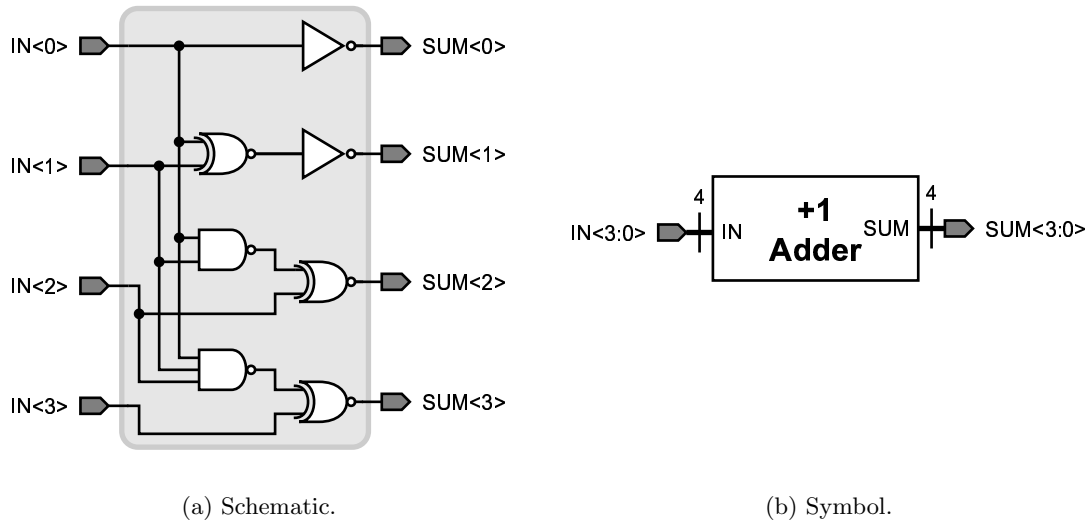
Figure 4-19: Integration controller 4–bit +1 digital adder.

(a) Schematic.   (b) Symbol.
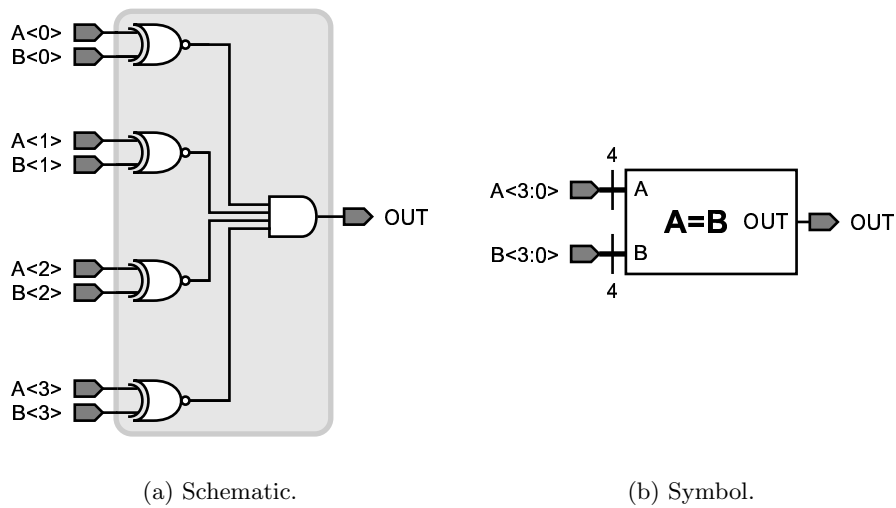
Figure 4-20: Integration controller 4–bit digital comparator.

by the external VCPL voltage. This scheme enables the use of different pixel reset types: a typical constant value reset level can be achieved by making PCOMP HIGH permanently and adjusting V_{VCPL} to the desired value. A pulsed pixel reset can be achieved by making PCOMP a clock waveform with the appropriate frequency and duty cycle.

A 4–bit digital multiplexer produces the new index to be stored in the associated SRAM location (Figure 4-21). If the pixel is reset, the new index is the time stamp, whereas if the

(a) Schematic.

(b) Symbol.
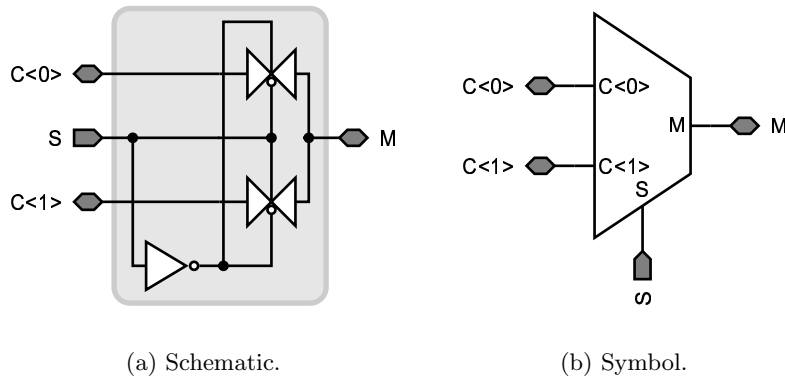
Figure 4-21: Integration controller digital multiplexer (only one channel shown).
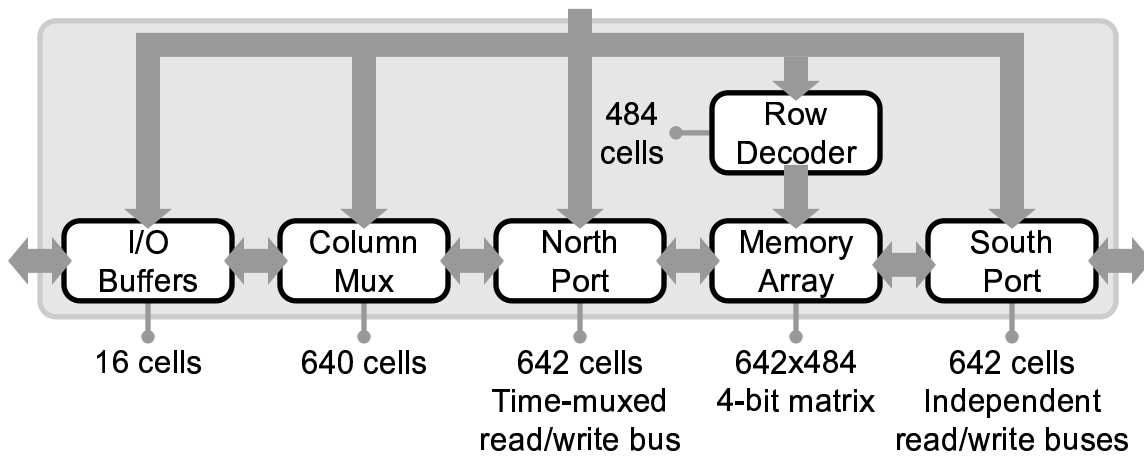
Figure 4-22: Memory block diagram.

pixel is not reset, the new index is simply the previously stored index.

4.5   Memory

Static random access memory (SRAM) cells store the index of the last reset cycle for every pixel in the sensing array. Access to the main 4–bit 642×484 array is provided by two ports: the North port, which is 8–bits wide with time–multiplexed input/output capabilities for external communication; and the South port, which is column–parallel with independent read and write terminals that connect with the column–parallel integration controller. A decoder provides memory row addressing capabilities (Figure 4-22).
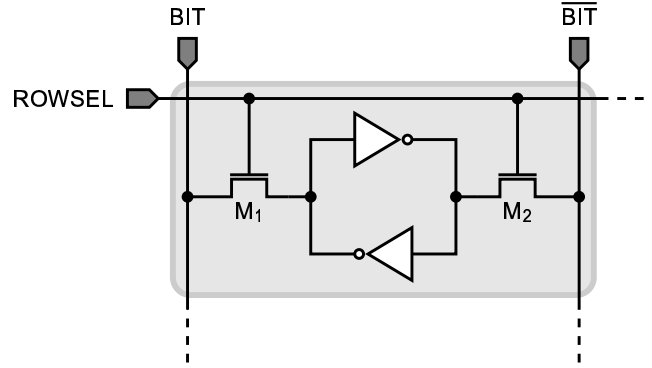
Figure 4-23: Memory cell schematic.

4.5.1   SRAM Cell

A fully static memory cell (Figure 4-23) was chosen for its robustness in an environment with a large number of photo–generated carriers, and for its relatively small area which enabled the layout of 4 cells pitch–matched to a single pixel width.

A pair of minimum–size digital inverters connected in a loop configuration store one bit of information. Read/write access from/to the complementary bit lines `BIT` and $\overline{\text{BIT}}$ is provided by transistors $M1$–$M2$ (which are also minimum size) and the row access line `ROWSEL`.

The bit lines have to be pre–charged to a mid–point voltage and then tri–stated before a read operation takes place. When the `ROWSEL` becomes active (HIGH) the outputs of the inverters are then connected to the bit lines. A differential voltage is eventually established in the these lines, but due to their large capacitance and the small driving capabilities of the inverters, a column sense amplifier is needed to speed up the process.

The bit lines are complementary held at the appropriate voltages (ground and V_{DD}) for a write operation. When the `ROWSEL` control line is HIGH the bit lines overwhelm the cell inverters charging or discharging its output nodes. When these nodes are safely past the inverters trip points, `ROWSEL` can become inactive isolating the cells and allowing the inverters to finish driving the internal nodes to the rails.

4.5.2   South Port

The sense amplifier in this block amplifies the small differential signal present in the bit lines during a read operation, establishes the correct voltages in the bit lines during
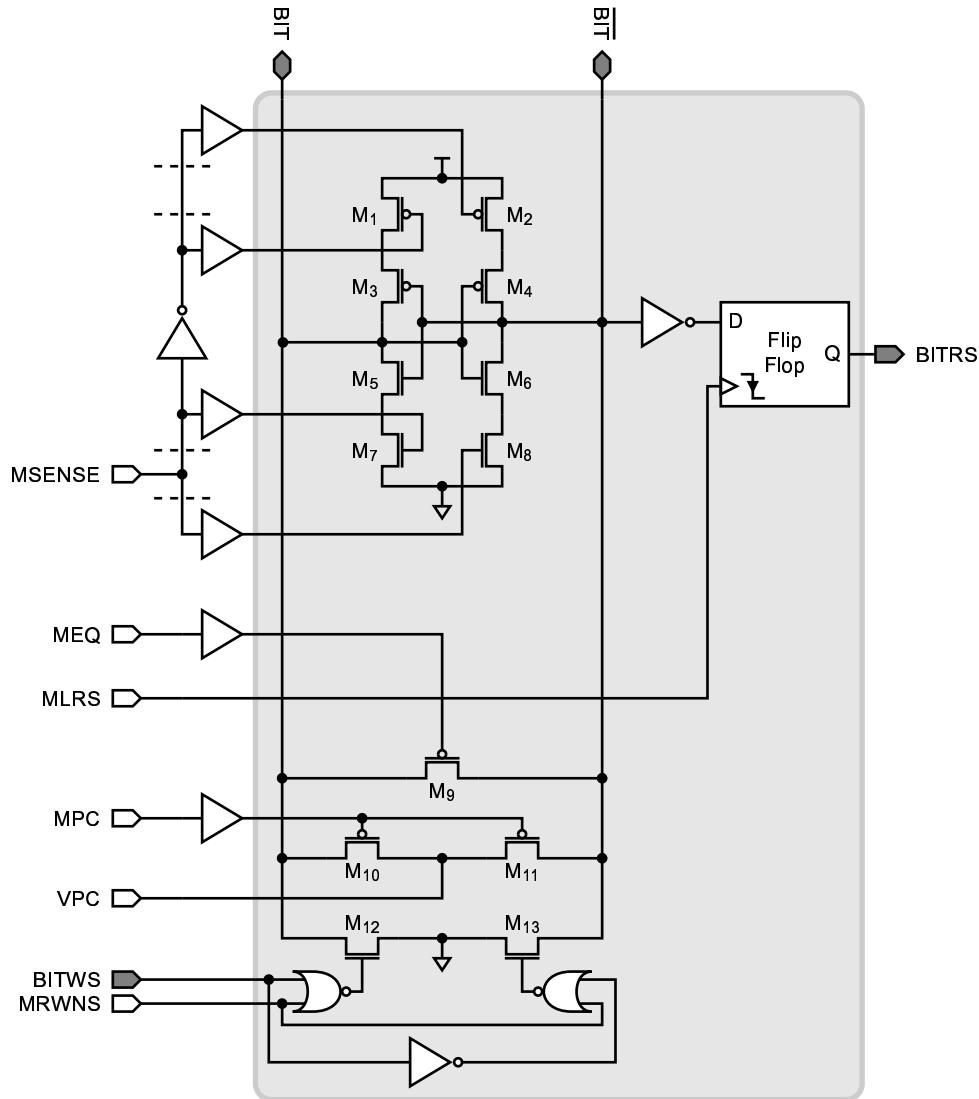
Figure 4-24: Memory South port schematic.

write operations and pre–charges the bit lines to a common potential when appropriate (Figure 4-24).

Transistors $M1$–$M8$ form a cross–coupled inverter pair activated when the MSENSE signal is HIGH. Transistors $M9$–$M11$ are identical, with a large aspect ratio. $M9$ equalizes both bit lines when MEQ is active, while $M10$–$M11$ pre–charge the bit lines to the external voltage VPC when MPC is active. This arrangement ensures that all sense amplifiers begin the signal amplification at the same time. If only $M9$ was present the equalized voltage would be column–dependent which can lead to read errors due to incomplete settling [8]. A D–type flip–flop stores the bit read on the falling edge of the MLRS signal. Transistors $M12$–$M13$ and
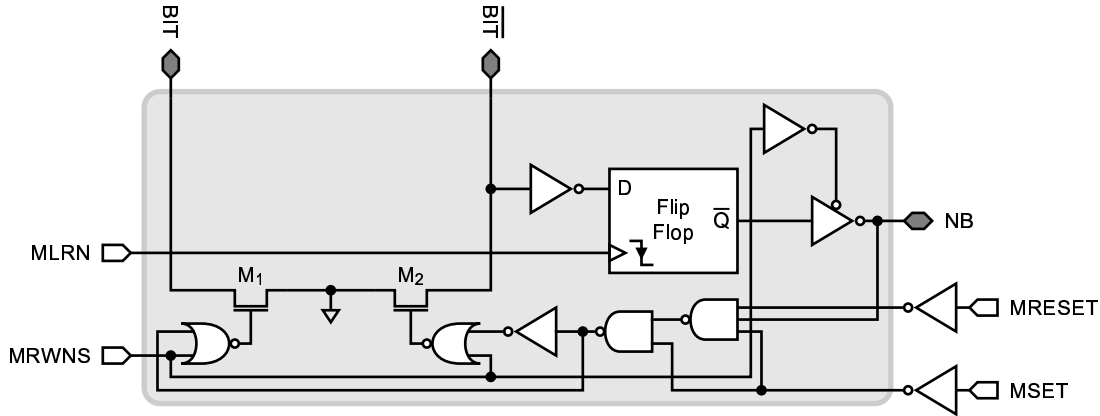
Figure 4-25: Memory North port schematic.

Table 4.2
SRAM North port global signals truth table

MRWNN	MSET	MRESET	NB	Bit to Write	Description
1	X	X	X	X	Read operation
0	1	X	X	1	Row set
0	0	1	X	0	Row reset
0	0	0	B	B	Write input bit

their associated logic ground either BIT or $\overline{\text{BIT}}$ depending on the value of BITWS, whose value
is provided by the integration controller. When MRWNS is LOW (read), BIT is grounded for
BITWS $= 0$, and $\overline{\text{BIT}}$ is grounded for BITWS $= 1$. A HIGH MRWNS indicates a write operation
thus $M12$–$M13$ are turned off regardless of the BITWS line.

4.5.3   North Port

This block is a paired–down version of the South port (Figure 4-25). An inverting
read buffer with tri–state capabilities controlled by the MRWNN signal was added to provide
adequate driving through the output multiplexer. Extra logic was also included to add
row–wise set/reset functionality. Provided MRWNN is LOW, when MSET is HIGH all cells in
the row addressed are written with a logic 1, while when MRESET is HIGH all cells in the
row addressed are written with a logic 0 (Table 4.2).

The signaling and timing for read and write operations is identical to those of the South
port (MLRN replaces MLRS, MRWNN replaces MRWNS). The main difference is that to achieve
true X–Y write capabilities a read operation has to precede a write operation. Since the
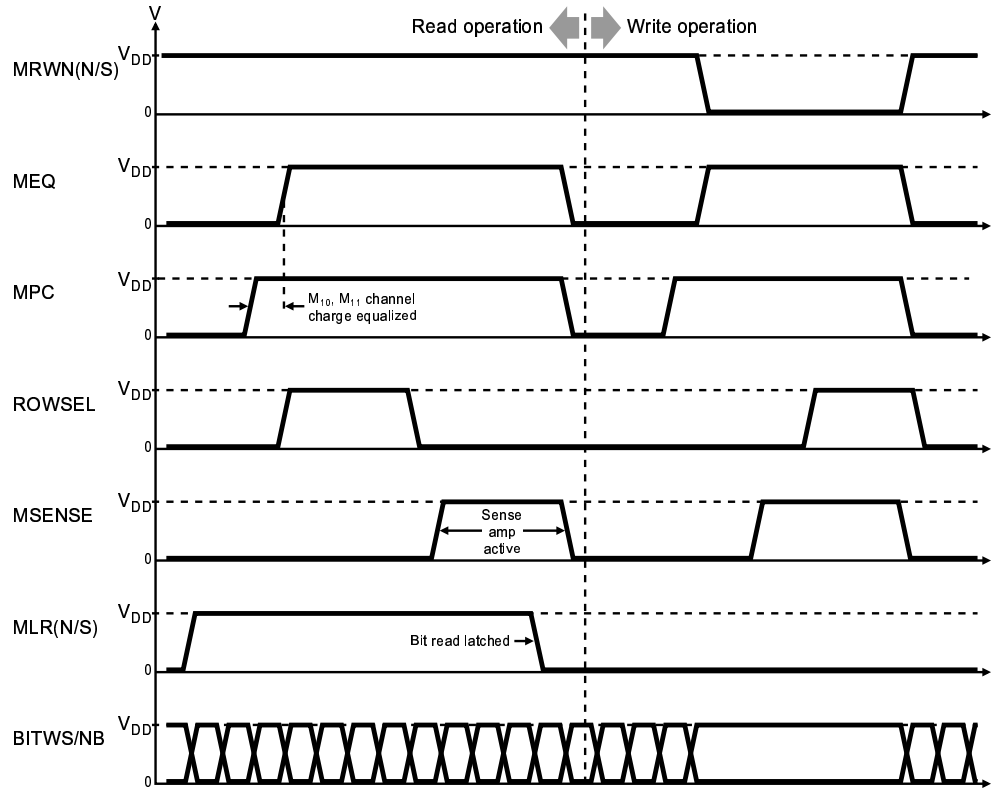
Figure 4-26: Memory phases for read and write operations.

input/output bus is 8–bits wide, only 2 4–bit words are explicitly driven at a time, but the rest of the columns in the row will be refreshed with the values of the last row read. These values are stored in node NB, which is isolated for write operations when its particular column is not addressed (the read buffer is tri–stated and the digital multiplexer is not driving the node). Therefore if two words of row $r \in [0, 483]$ need to be written, row r has to be read and immediately updated to avoid an unintentional partial row copy. The North port write operation thus takes more time to complete than other memory I/O operations; however this feature is only added for circuit testing and debugging purposes.

4.5.4   Phases of Operation

Read

This operation begins with the bit lines equalized and pre–charged to VPC, consequently both MEQ and MPC are LOW (Figure 4-26). The appropriate ROWSEL line is activated when the row address is input to the decoder and the cells from the selected row start to create

a differential voltage in the bit lines. Once this voltage is larger than the comparator offset the row decoder can be deactivated (`ROWSEL` LOW) and the dynamic comparator can be enabled (`MSENSE` HIGH). After the bit lines have settled to its rail values the read bit can be latched in the D–type flip–flop by lowering `MLRS` (South port) or `MLRN` (North port).

Write

The bit lines also start equalized for this operation. When the information to be written is ready and present in the `BITRS` (South port) or `NB` (North port) line, `MRWNS` (South port) or `MRWNN` (North port) is lowered to indicate a read operation and to ground the appropriate bit line. The dynamic comparator is then activated with the `MSENSE` line to drive the bit lines to the rails (if not one line would be at ground and the other at `VPC`). Once this is achieved the row decoder can be activated and the driven bit lines overpower the inverters in the cell thus writing the desired bits.

4.5.5   Column Multiplexer and Input/Output Buffers

To reduce the total IC pin count the memory external communication is limited to 8–bits or 2 words with a digital multiplexer. A standard pass–gate chain design was implemented (Figure 4-27). The binary–encoded column information is expected in the `MCOL<8:0>` bus and is latched into a 9–bit D–type flip–flop register on the negative edge of `MCMLATCH`. A 1–to–2 decoder generates the `MC<i>` and $\overline{\text{MC<i>}}$ signals (i $\in [0, 8]$) that run through the multiplexer being connected to the regular or inverted input of the pass gates.

The rightmost and leftmost memory columns are not multiplexed and run directly to the input/output buffer, they can be accessed on the `MIOTL<3:0>` and `MIOTR<3:0>` buses respectively. For the other columns, there are $642 \times 4 = 2568$ bit lines, and the connection to the pass gates is as follows:

$$\text{MC<i> connected to} \quad \begin{matrix} \text{regular} \\ \text{inverted} \end{matrix} \quad \text{NB}_j \text{ bit line pass–gate input if} \quad \begin{matrix} \left\lfloor \frac{j}{8 \cdot (i+1)} \right\rfloor \; mod \; 2 = 1 \\ \left\lfloor \frac{j}{8 \cdot (i+1)} \right\rfloor \; mod \; 2 = 0 \end{matrix}$$

where j $\in [0, 2567]$. Consequently pairs of adjacent columns are addressed by `MCOL<8:0>`. The leftmost memory column of the addressed pair is accessed in the `MIOB0<3:0>` bus while the rightmost column is accessed in the `MIOB1<3:0>`.
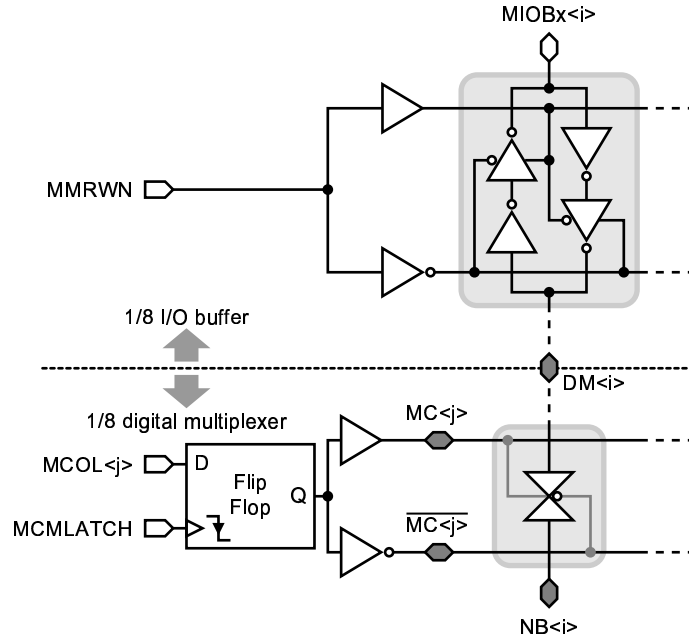
Figure 4-27: Memory column multiplexer and input/output buffer schematic.

The relationship between the multiplexer lines DM_j, $j \in [0, 2568]$ and the output buses is the following:

MIOB0<0>		$j \bmod 8 = 0$	MIOB1<0>		$j \bmod 8 = 4$
MIOB0<1>	connected to j:	$j \bmod 8 = 1$	MIOB1<1>	connected to j:	$j \bmod 8 = 5$
MIOB0<2>		$j \bmod 8 = 2$	MIOB1<2>		$j \bmod 8 = 6$
MIOB0<3>		$j \bmod 8 = 3$	MIOB1<3>		$j \bmod 8 = 7$

The memory has 16 input/output non–inverting buffers, whose direction is controlled by the MMRWNN signal (Figure 4-27). MMRWNN HIGH indicates a read (output) operation while a MMRWNN LOW signal indicates a write (input) operation.

4.5.6   Row Decoder

The 4–bit memory cell is 5.78μm high, shorter than the pixel height. Consequently the same area limitations that led to the pre–decoding scheme in the pixel row decoder apply to the memory row decoder. However, since no simultaneous access to multiple rows is necessary, most of the addressing functionality can be moved to the pre–decoder, which is shown in Figure 4-28.

The binary–encoded memory row address is expected in the MROW<8:0> bus and is
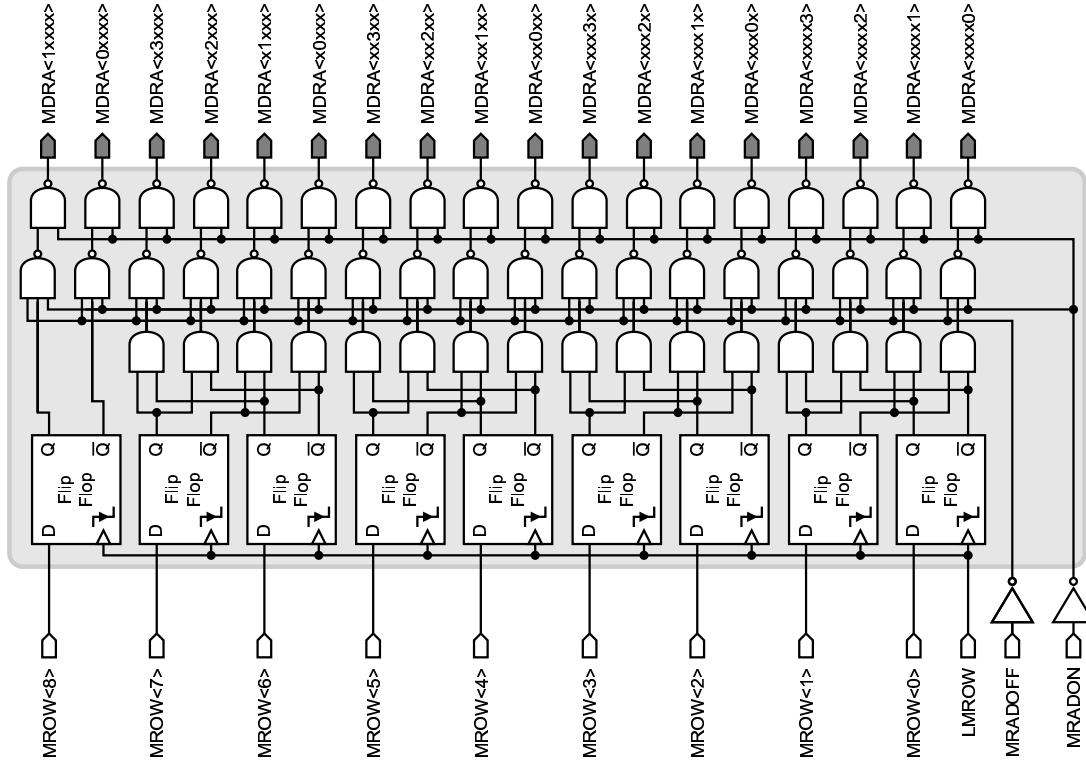
Figure 4-28: Memory row decoder pre–decoder schematic.

Table 4.3

Memory row decoder global signals truth table

MRDAON	MRDAOFF	Description
0	0	Single memory row activation
1	X	All memory rows active
0	1	All memory rows inactive

latched by a register of D–type flip–flop latches on the negative edge of LMROW. A series
of 2–to–4 decoders form five groups in which only one line is active per addressed row.
Additional logic follows to provide global on/off capabilities. When MRDAON is HIGH all
lines in all groups are active, and if MRDAOFF is HIGH all lines in all groups are inactive
(Table 4.3).

The row address decoder core (shaded area), which is repeated at every pixel row,
consists only of logic for the final address decoding process (Figure 4-29). A 4–input AND
gate takes input from one line out of each of the five most significant groups (generated by
MROW<8:2>) and thus selects four pixel rows. The final selection is made by the 2–input
AND gate, whose inputs are the output of the 4–input AND gate and one line out of the
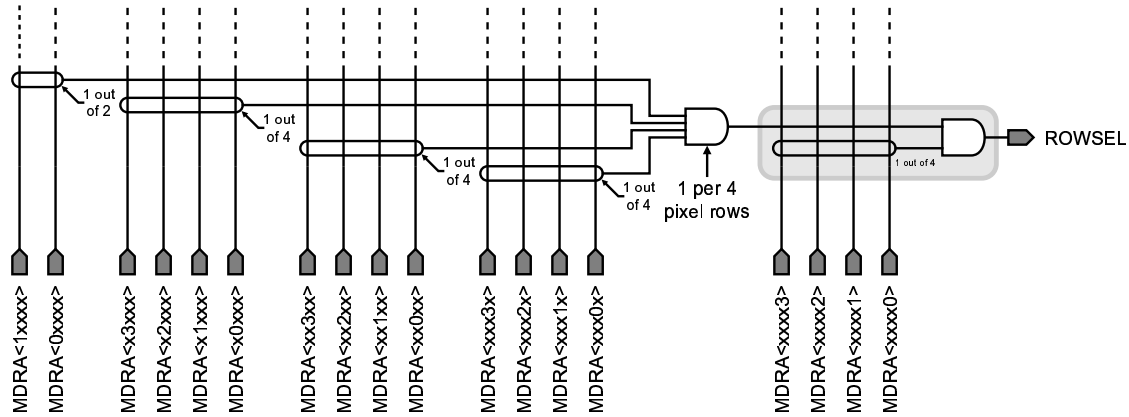
Figure 4-29: Memory row decoder schematic (row circuitry).
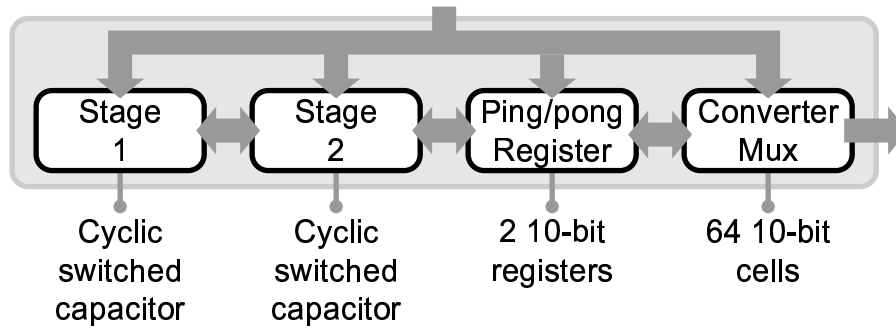
Figure 4-30: Analog–to–digital converter block diagram.

least significant group (generated by `MROW<1:0>`).

4.6   Analog–to–Digital Converter/Correlated Double Sampler

The ADC/correlated double sampler (CDS) vector is made of 64 cells. Each stage has two modes of operation: conventional ADC or correlated double sampling. In ADC mode the cells are configured as 2–stage cyclic converters whose results are stored in 10–bit ping–pong registers (Figure 4-30). A 10–bit wide decoder selects one ADC and routes its conversion result to the output bus. For testing and debugging purposes, the ADC can convert an external differential voltage carried by the `AEIP` and `AEIN` lines ($V_{ADC_{IN}} = V_{AEIP} - V_{AEIN}$) when the `IESEL` control signal is LOW (Figure 4-31).

The cyclic process produces one bit of resolution per cycle. It first determines if the stage input voltage is in the upper or lower half of the differential input range (-2V to
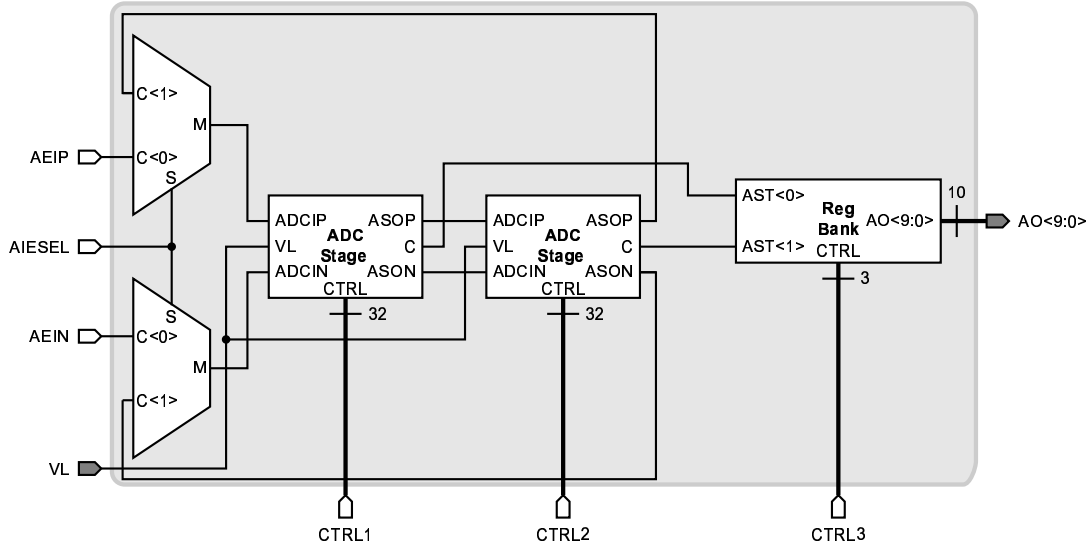
Figure 4-31: Analog–to–digital converter schematic.

+2V in the experimental chip) generating a digital 1 or 0 respectively. The half where the input voltage was determined to be in is then expanded to fill the full input range. This operation, known as residue generation, can be expressed mathematically as:

$$V_{RES} = 2 \cdot V_{IN} \pm V_{REF} \tag{4.8}$$

where $V_{REF} = 2$V. This reference voltage is subtracted if the stage bit is a digital 1, and added if the stage bit is a digital 0. The conversion process can then be repeated again to obtain a new bit of resolution, thus generating bits from most significant (MSB) to least significant (LSB).

The CDS mode reduces fixed pattern noise by subtracting the pixel reset level from the pixel output at the end of the integration time. As a by–product, the single–ended pixel output is converted into a scaled differential signal. Both stages are capable of performing the CDS operation with the multiplexed pixel output lines that connected to the VL ADC/CDS input.

A circuit that has the needed functionality is shown in Figure 4-32. It consists of 6 capacitors, a fully differential operation amplifier (opamp), a dynamic comparator and pass gates. Capacitors $CA1T$, $CA1B$ are used in ADC mode, capacitors $CC1T$, $CC1B$ are used in CDS mode, and capacitors $C2T$, $C2B$ are used in both modes. The phases of operation for one CDS cycle and a 2–bit conversion are shown in Figure 4-33.
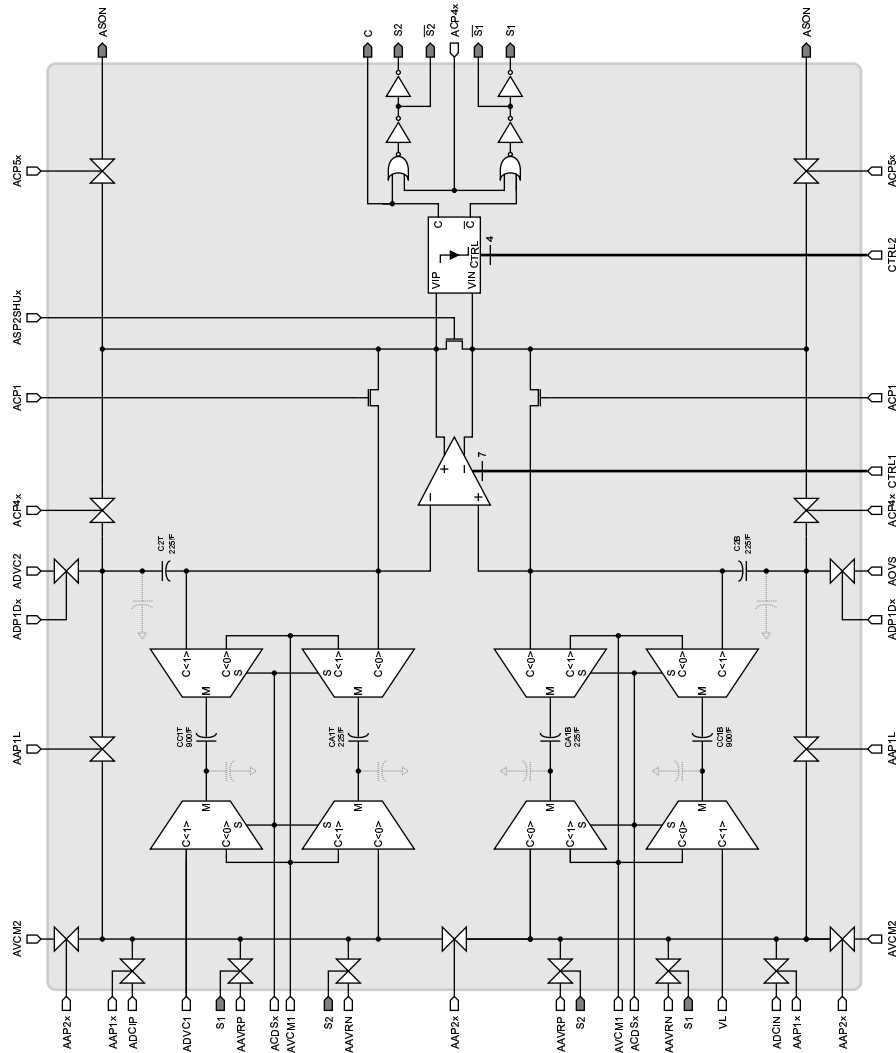
(a) Schematic. Inverted inputs of pass gates not indicated for simplicity. All external signals are level converted to $V_{EE} = 3.3$V. $x \in \{0, 1\}$.
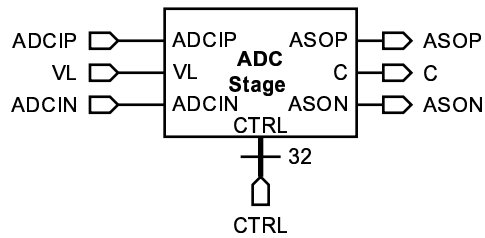
(b) Symbol.

Figure 4-32: Analog–to–digital converter cyclic stage/correlated double sampling stage.
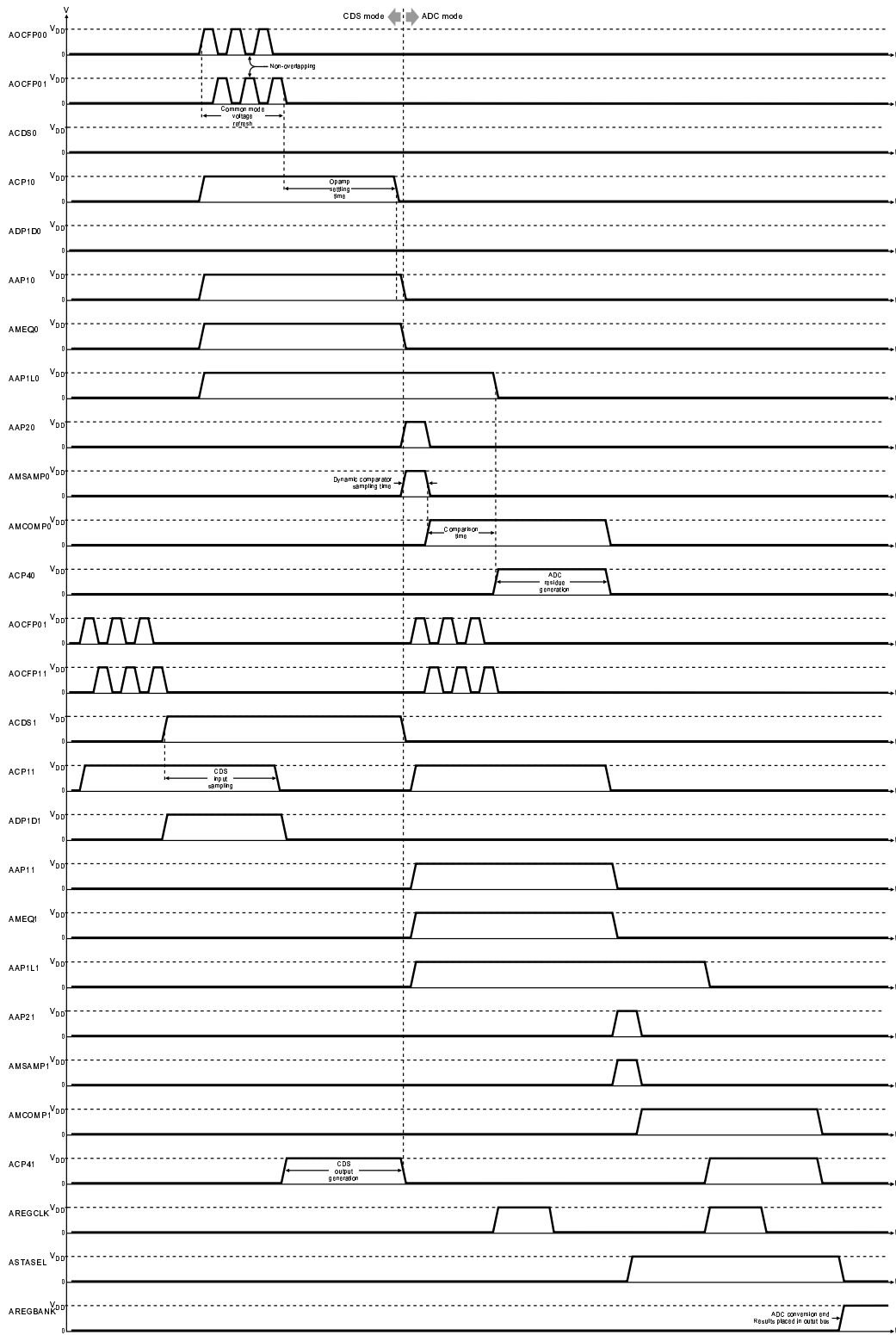
Figure 4-33: Analog–to–digital converter and correlated double sampling phases of operation. x $\in \{0, 1\}$.
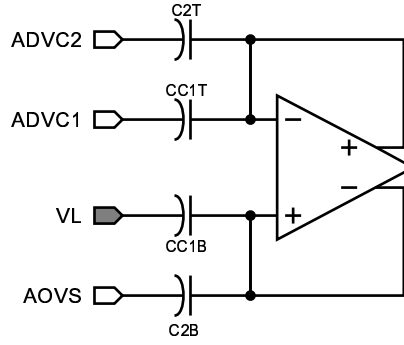
Figure 4-34: Stage configuration during the correlated double sampling input voltage sample phase.

4.6.1 CDS Mode

The pixel output immediately after the end of the integration time is read during the first phase of operation in CDS mode. `ACDSx`, `ADP1Dx` and `ACP1x` are all HIGH, putting the opamp in unity gain feedback while also sampling the input (Figure 4-34). For the experimental chip, $V_{ADCV1} = V_{ADCV2} = 0.65\text{V}$ and $V_{AOVS} = 2.65\text{V}$. Assuming very high opamp differential gain and matched capacitors ($CC1T \equiv CC1B = CC_1$ and $C2T \equiv C2B = C_2$), the charge at the opamp input terminals is:

$$
\begin{aligned}
q^-(t_1) &= CC_1 \cdot \left(V_L(t_1) - V_O^-(t_1)\right) + C_2 \cdot \left(V_{AOVS} - V_O^-(t_1)\right) \\
q^+(t_1) &= CC_1 \cdot \left(V_{ADVC1} - V_O^+(t_1)\right) + C_2 \cdot \left(V_{ADVC2} - V_O^+(t_1)\right)
\end{aligned}
\tag{4.9}
$$

Labeling $V_{CM} = V_O(t_1) = V_O^+(t_1) - V_O^-(t_1)$ then:

$$
\Delta q(t_1) = CC_1 \cdot (V_{ADVC1} - V_{CM} - V_L(t_1)) + C_2 \cdot (V_{ADVC2} - V_{CM} - V_{AOVS}) \tag{4.10}
$$

The pixel reset level is read in the second and last phase of operation in CDS mode. Here `CDSx` and `ACP4x` are high resulting in the configuration shown in Figure 4-35. The charge at the opamp input terminals is:

$$
\begin{aligned}
q^-(t_2) &= CC_1 \cdot \left(V_L(t_2) - V_O^-(t_1)\right) + C_2 \cdot \left(V_O^-(t_2) - V_O^-(t_1)\right) \\
q^+(t_2) &= CC_1 \cdot \left(V_{ADVC1} - V_O^+(t_1)\right) + C_2 \cdot \left(V_O^+(t_2) - V_O^+(t_1)\right)
\end{aligned}
\tag{4.11}
$$

Then:

$$
\Delta q(t_2) = CC_1 \cdot (V_{ADVC1} - V_{CM} - V_L(t_2)) + C_2 \cdot (V_O(t_2) - V_{CM}) \tag{4.12}
$$
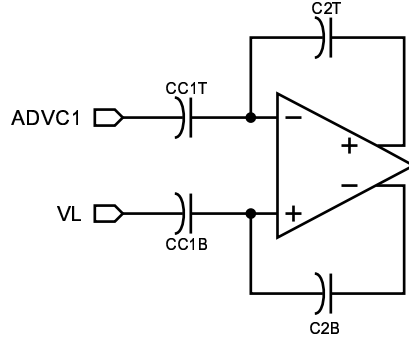
Figure 4-35: Stage configuration during the correlated double sampling output phase.

Charge conservation demands $\Delta q\left(t_1\right) = \Delta q\left(t_2\right)$ thus:

$$V_O\left(t_2\right) = V_O^+\left(t_2\right) - V_O^-\left(t_2\right) = \frac{CC_1}{C_2} \cdot \left(V_L\left(t_2\right) - V_L\left(t_1\right)\right) + \left(V_{ADVC1} - V_{AOVS}\right) \quad (4.13)$$

In the experimental chip $CC_1 = 900\text{fF}$ and $C_2 = 225\text{fF}$ nominally[2], so with $\Delta V_L = V_L\left(t_2\right) - V_L\left(t_1\right)$:

$$V_O\left(t_2\right) \approx 4 \cdot \Delta V_L - 2 \quad (4.14)$$

Hence ΔV_L is the pixel signal, that is, the pixel output change during the integration time. Also, since the pixel output range is nominally $[0.25\text{V}, 1.25\text{V}]$ single–ended, the CDS converts it to a $[-2\text{V}, +2\text{V}]$ differential range.

4.6.2   ADC Mode

As in CDS mode, the input voltage is sampled during the first phase of operation in ADC mode. `ACP1x`, `AAP1x`, `AAP1Lx` and `AMEQx` are HIGH, putting the opamp in the configuration shown in Figure 4-36. The charge at the opamp input terminals is ($CA1T \equiv CA1B = CA_1$ assumed):

$$\begin{aligned}
q^-\left(t_1\right) &= CA_1 \cdot \left(V_{ADCIN} - V_O^-\left(t_1\right)\right) + C_2 \cdot \left(V_{ADCIN} - V_O^-\left(t_1\right)\right) \\
q^+\left(t_1\right) &= CA_1 \cdot \left(V_{ADCIP} - V_O^+\left(t_1\right)\right) + C_2 \cdot \left(V_{ADCIP} - V_O^+\left(t_1\right)\right)
\end{aligned} \quad (4.15)$$

---

[2]Stage capacitor magnitudes mainly determined by a trade–off between settling time, mismatch tolerances and noise to achieve the desired ADC/CDS performance.
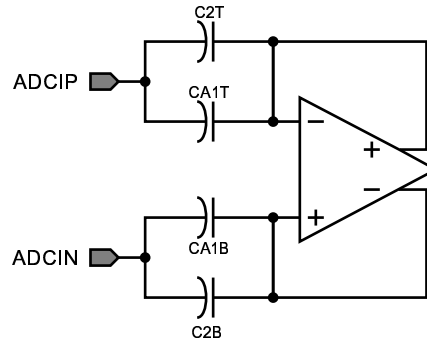
Figure 4-36: Stage configuration during the analog–to–digital converter sample phase.
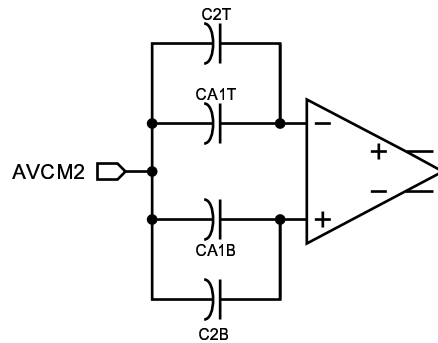
Figure 4-37: Stage configuration during the analog–to–digital converter comparison phase.

Defining $V_{IN} = V_{ADCIP} - V_{ADCIN}$:

$$\Delta q\left(t_1\right) = CA_1 \cdot \left(V_{IN} - V_{CM}\right) + C_2 \cdot \left(V_{IN} - V_{CM}\right) \tag{4.16}$$

The stage bit is determined during the second phase of operation in ADC mode, the comparison phase. Here `AAP1Lx`, `AAP2x` and `AMSAMPx` are HIGH, which results in the configuration shown in Figure 4-37. The opamp is used in this case as a offset–compensated pre–amplifier for the dynamic latch. Running open loop, the differential output of the opamp increases rapidly in the direction of the input voltage sign (positive or negative). This voltage difference need not be significant, it only needs to be larger than the comparator offset, so the pre–amplification phase can be of short duration. The `AAP2SHUx` signal can optionally be turned HIGH after the comparison has taken place but before the residue is calculated. This signal shorts the opamp outputs and thus helps in making a faster transition be-
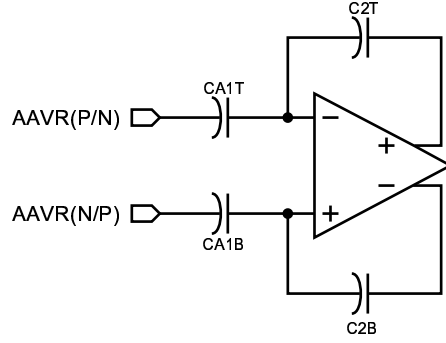
Figure 4-38: Stage configuration during the analog–to–digital converter output phase.

tween the pre–amplification and output modes of the opamp, specially if as a result of the pre–amplification its outputs are close to the rails.

Once the comparison takes place the third and last phase of operation in ADC mode, the output/residue generation phase, begins. `AMCOMPx` and `ACP4x` are HIGH, resulting in the configuration shown in Figure 4-38. The charge at the opamp inputs is:

$$
\begin{aligned}
q^- (t_2) &= CA_1 \cdot \left(V_{AAVR(N/P)} - V_O^- (t_1)\right) + C_2 \cdot \left(V_O^- (t_2) - V_O^- (t_1)\right) \\
q^+ (t_2) &= CA_1 \cdot \left(V_{AAVR(P/N)} - V_O^+ (t_1)\right) + C_2 \cdot \left(V_O^+ (t_2) - V_O^+ (t_1)\right)
\end{aligned}
\tag{4.17}
$$

Then with $V_{AAVR} = V_{AAVRP} - V_{AAVRN}$:

$$
\Delta q (t_2) = CA_1 \cdot \left(\pm V_{AAVR} - V_{CM}\right) + C_2 \cdot \left(V_O (t_2) - V_{CM}\right)
\tag{4.18}
$$

Charge conservation demands $\Delta q (t_1) = \Delta q (t_2)$ thus:

$$
V_O (t_2) = V_O^+ (t_2) - V_O^- (t_2) = \frac{CA_1 + C_2}{C_2} \cdot V_{IN} \pm \frac{CA_1}{C_2} \cdot V_{AAVR}
\tag{4.19}
$$

Nominally $CA_1 = C_2 = 225\text{fF}$, and labeling $V_{RES} = V_O (t_2)$ and $V_{REF} = V_{AAVR} = 2\text{V}$:

$$
V_{RES} = 2 \cdot V_{IN} \pm 2
\tag{4.20}
$$

which matches the desired behavior of the stage (Equation 4.8). If the stage bit is a digital 0, $+V_{REF}$ is added to the scaled input, while $-V_{REF}$ is added if the stage bit is a digital 0. This add/subtract operation can easily be achieved by flipping the differential terminals of
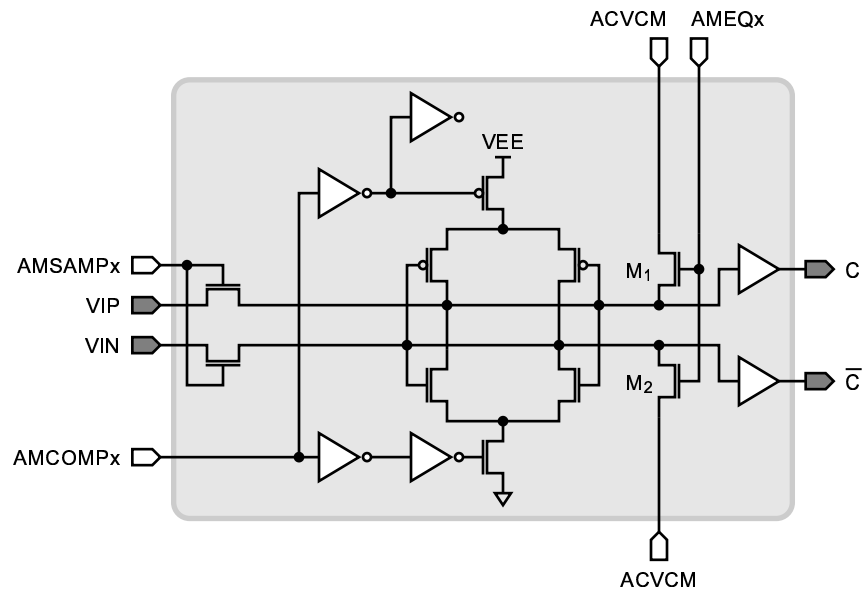
(a) Schematic.
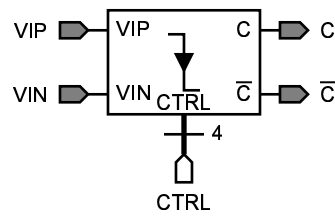
(b) Symbol.

Figure 4-39: Analog–to–digital converter cyclic stage comparator.

the `AAVR(N/P)` input according to the stage bit, which is achieved by the `S1` and `S2` internal signals.

Comparator

The stage comparator is a dynamic comparator similar to the one used in the integration controller (Figure 4-39). Transistors $M1$ and $M2$ allow for the comparator internal nodes to be equalized to the voltage of the `ACVCM` line, V_{ACVCM}. When the comparator is not in use the `MEQx` line is always HIGH which ensures fast comparator recovery and sampling.

The opamp outputs are sampled when `AMSAMPx` is HIGH and the cross–coupled inverters
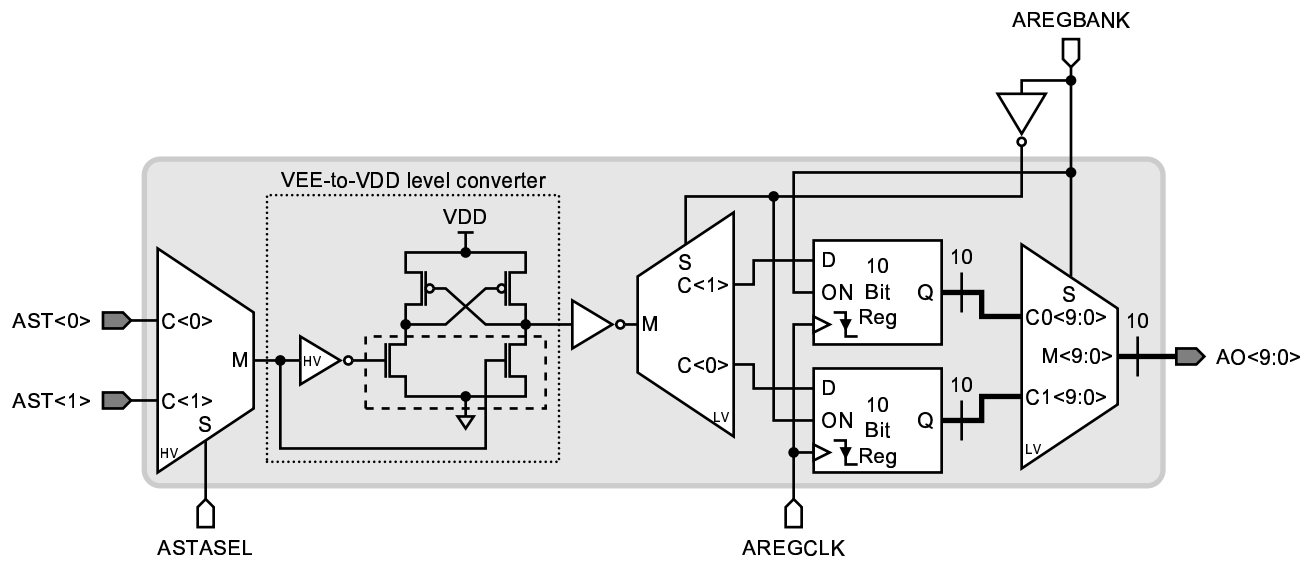
(a) Schematic.
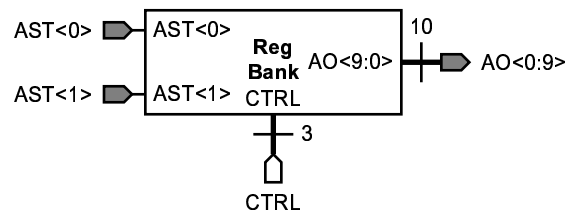
(b) Symbol.

Figure 4-40: Analog–to–digital converter ping–pong register.

are activated when the `AMCOMPx` line is HIGH. The comparator output is not stored in a flip–flop thus `AMCOMPx` must remain active for as long as the comparison result is needed, that is, for as long as `ACP4x` is HIGH and the opamp outputs are settling to the residue voltage.

4.6.3 Ping–pong Register

The stage bits need to be stored with the appropriate sequence in a shift register. A circuit with this functionality is shown in Figure 4-40. A multiplexer controlled by the `ASTASEL` line selects the comparator output of one of the stages. This signal then goes through a 3.3V–to–1.8V level converter and another multiplexer controlled in this case by
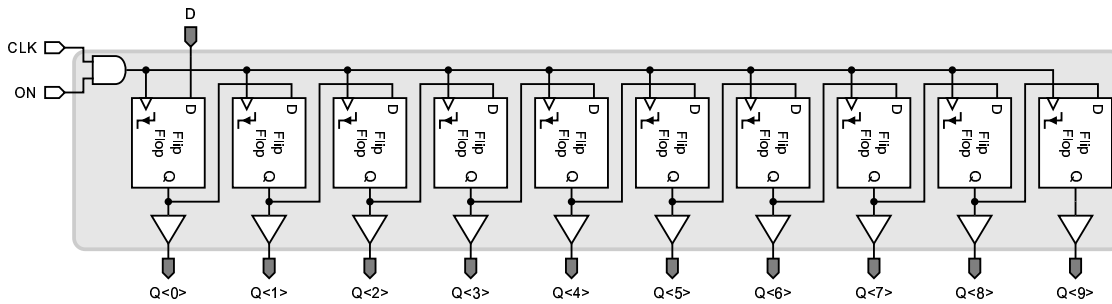
(a) Schematic.
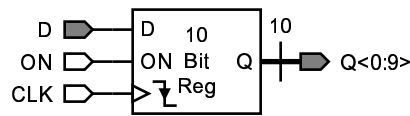
(b) Symbol.

Figure 4-41: Analog–to–digital converter shift register.

the `ADCREGBANK` line which selects one of two shift registers clocked by the `AREGCLK` line to store the bits in. A 10–bit 2–channel multiplexer also controlled by the `ADCREGBANK` line selects the register that is not in use by the ADC for output so the results of the last conversion can be read while the converter is performing the next quantization.

4.6.4 Shift Register

The shift register design used by the ADC ping–pong register is shown in Figure 4-41. It is made of 10 negative edge D–type flip–flop stages with a clock enable feature. When `ON` is LOW the clock is disabled, but to avoid spurious latching `ON` has to become inactive when `CLK` is also inactive.

4.6.5 Output Multiplexer

The ADC output bus is only 10–bits wide so a 10–bit 64 channel multiplexer is required between the ADC vector and the IC exterior (Figure 4-42). The binary–encoded ADC number is expected in the `ACOL<5:0>` bus, which is latched on the negative edge of `ALCOL`.
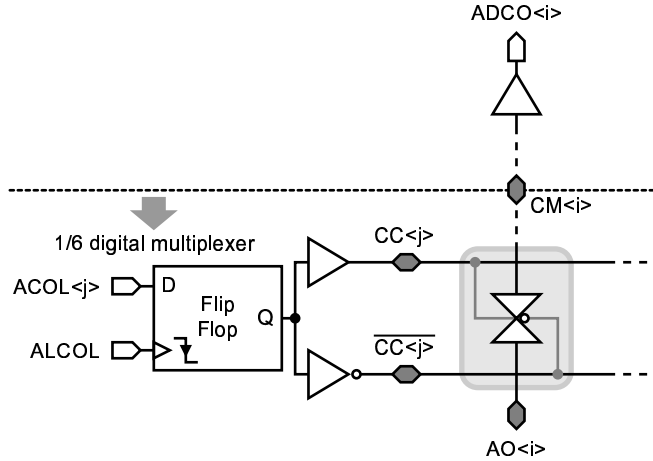
Figure 4-42: Analog–to–digital converter output multiplexer.

4.6.6 Operational Amplifier

The opamp topology used in the ADC/CDS stages is shown in Figure 4-43. It is a classic PMOS input pair cascoded two–stage design with a capacitive–resistive compensation network (Z_C). The circuit requires three external references: a current `AOIBIASx`, nominally 100μA, and voltages `AOVREF`, `AOVCM`, nominally $V_{EE}/2 = 1.65$V.

Transistors $MSU1$–$MSU3$ form a small start–up circuit whose mission is to ensure that $MB2$ is on soon after the power supply reaches its quiescent value. The bias point avoided is a voltage close to V_{EE} at $MB2$'s gate. Transistor $MSU3$ lowers the potential of $MSU2$'s gate until this transistor, which has a large aspect ratio, turns on significantly pulling down $MB2$'s gate. This helps the external reference current establish the right currents (and therefore voltages) across the different current mirrors of the bias circuit. When $MSU2$ turns on $MSU1$ also, eventually, turns on charging up the gate node of $MSU2$ as its current drive overwhelms $MSU3$. This process quickly turns transistor $MSU2$ off and after that only a small current flows through the now disabled start–up circuitry.

Transistors $MB1$–$MB20$ form the bias circuit. It features a PMOS and NMOS high compliance cascoded current mirrors to produce the five bias voltages (V_{B_1}–V_{B_5}) needed by the active (signal) part of the opamp. Each bias leg draws 10μA of current.

Transistors $M1$–$M16$ form the active part of the opamp. $M1$–$M2$ implement the tail current for the differential pair $M3$–$M4$ which is actively loaded by transistors $M9$–$M10$. Both differential pair and active load are cascoded with $M5$–$M8$ to increase the stage output
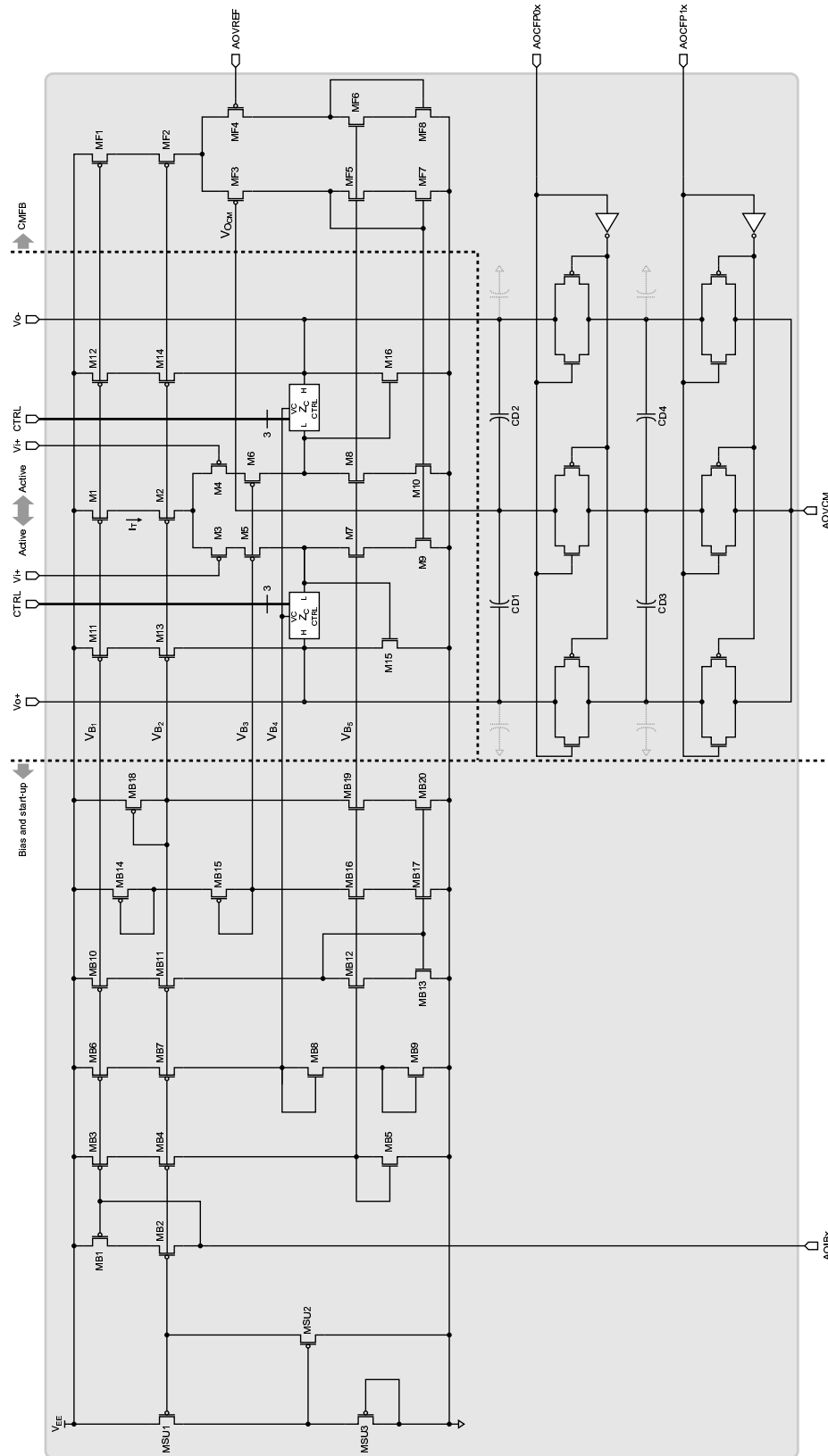
Figure 4-43: Operational amplifier used in the analog–to–digital converter stages.

resistance. Transistors $M15$–$M16$ form the common–source second stage, with cascoded active loads provided by $M11$–$M14$.

Transistors $MF1$–$MF8$ and capacitors $CD1$–$CD4$ form the common mode feedback (CMFB) of the opamp. $CD1$–$CD4$ are in a switched–capacitor circuit which nominally produces $V_{CM} = \left(V_O^+ + V_O^-\right)/2$. The midpoint of $CD1$–$CD2$ is a high impedance node, so its voltage has to be periodically refreshed to `AOVCM` by the `AOCFP0x` and `AOCFP1x` phases (Figure 4-33). A scaled–down version of the active differential pair inputs the voltage of the `AOVREF` line and the calculated opamp output common mode. The current through one of the legs of this CMFB differential pair is mirrored to the active loads of the opamp first stage, so when the measured common mode differs from the reference, the current flowing through transistor $MF7$ changes, modifying the first stage output voltage in the direction that eventually brings the opamp output common mode back toward the reference.

Compensation Network

The opamp is compensated using the dominant pole method which adds a capacitor between the outputs of the two amplifying stages. A resistance was added to further aid stability. Its actual implementation is an NMOS transistor whose gate is at potential V_{B_4} generated by the bias circuit (Figure 4-44).

An extra, smaller, capacitance was included in the compensation network to speed up the opamp response when it is being used as an offset–compensated pre–amplifier. Two analog multiplexers controlled by the `AOPAPx` external signal select which capacitor is in use at any point in time: when `AOPPAx` is LOW capacitor $CCM = 450\text{fF}$ is active, when `AOPPAx` is HIGH capacitor $CCP = 230\text{fF}$ is active. The nominal phase margin attained with CCM is $60°$, which is reduced to $45°$ with CCP. During the pre–amplification phase the input differential pair is saturated, that is to say, all the tail current I_T flows through one leg, and the differential output evolution can be approximated to be:

$$V_O\left(t\right) \approx \frac{I_T}{C_C} \cdot t \tag{4.21}$$

Consequently a smaller compensation capacitor yields a faster response time to reach a particular output voltage difference. Since $CCP \approx CCM/2$, the pre–amplification time is almost cut in half when this capacitor is used.
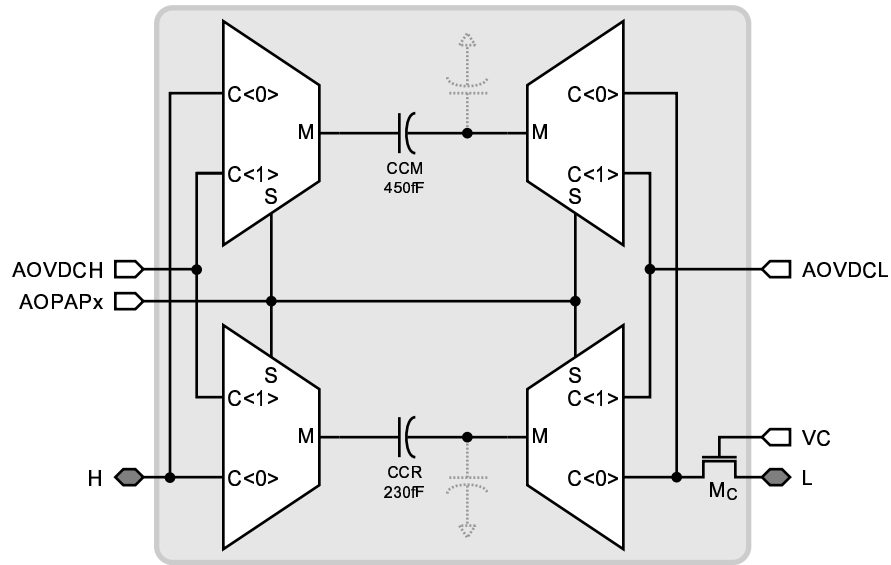
(a) Schematic.
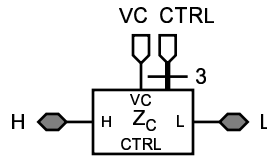
(b) Symbol.

Figure 4-44: Analog–to–digital converter operational amplifier frequency compensation network.

4.7 Summary

The characteristics, features and design of a proof–of–concept integrated circuit have been presented. The IC includes the following blocks:

- A VGA (640×420) sensing array. Pixels are squares 7.5μm on the side with a n^+–p substrate photodiode that occupies 49% of the pixel area. A 5 NMOS transistor design provides electronic shutter and conditional reset capabilites.

- An SRAM array capable of storing 4 bits per pixel and with dual ports for internal and external communication.

- An integration controller array that is fully column–parallel which implements the key

functionality required by the multiple sampling algorithm.

- An analog multiplexer that routes the three outputs lines present for every pixel column to the ADCs.

- A 64–element, 10–bit ADC/CDS array.

Chapter 5

Experimental Results

5.1   Test Setup

The setup used to characterize the proof–of–concept IC is shown in Figure 5-1. A custom test–board includes the necessary voltage references, bias currents and test structures (like a fully differential digital–to–analog converter to test the chip ADCs). The digital control for the IC was implemented in a field programmable gate array (FPGA), which also controlled the data communication between the test–board and an x86 computer. The LabVIEW software package [68] was used on the computer for data acquisition and initial post–processing. Further post–processing was done using Matlab [60].
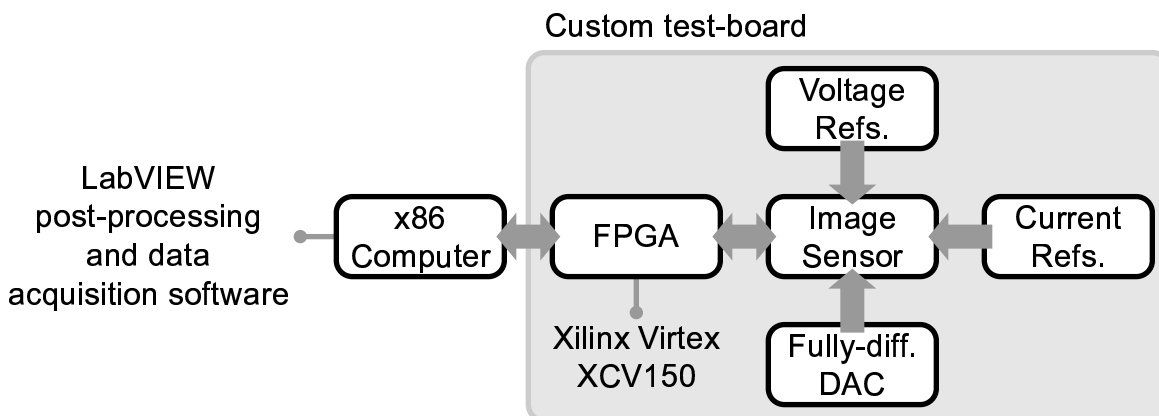
Figure 5-1: Block diagram of the test setup used to characterize the proof–of-concept integrated circuit.
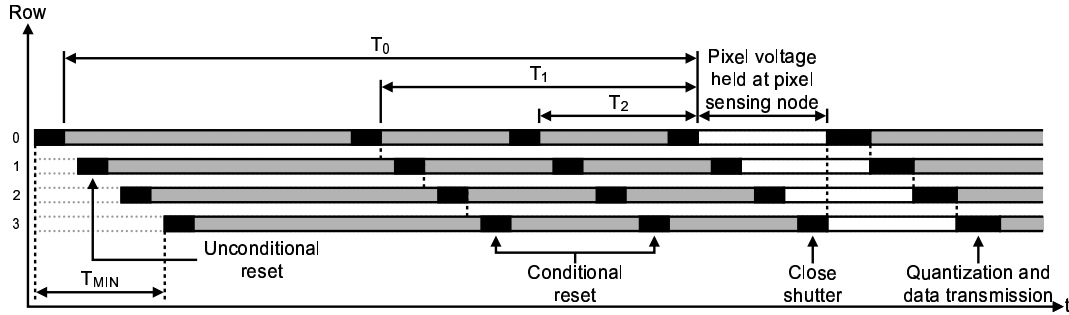
Figure 5-2: Light integration timing when the rolling shutter scheme is used. Pixel shutter transistor used to allow slow test board/computer data transmission.

5.2 Digital Control

The proof–of–concept integrated circuit only includes some basic digital blocks (decoders, multiplexers, etc.) and the critical elements of the integration controller. The rest of the digital control, including the light integration scheme, was implemented in a Xilinx Virtex XCV150 [69].

A rolling shutter scheme is the most common among integrating image sensors (Figure 5-2). Since some operations (conditional reset and quantization) use the same pixel output lines, rows cannot be processed in parallel. These operations are then applied in cycles, one row at a time, and consequently the integration periods of the different rows are "staggered"in time. The integration time starts when the rows are sequentially reset unconditionally. Several conditional reset cycles follow, and the integration time ends with a quantization cycle. The time shift between integration times of adjacent rows equals the time it takes to perform the slowest row operation. Namely, if the unconditional reset takes T_{UR} seconds, the conditional reset takes T_{CR} seconds, and the quantization takes T_{ADC} seconds, then the time shift T_{TS} is:

$$T_{TS} = max\left(T_{UR}, T_{CR}, T_{ADC}\right) \tag{5.1}$$

The integration controller does not provide a simple method to unconditionally reset the pixels. Therefore a special conditional pixel reset was used to perform this operation. In this particular instance the pixel output was not accessed, rather the appropriate $\texttt{CSEQC}_i$ line (Figure 4-10) was kept high so that the integration controller comparison was performed

between the pixel line equalization voltage (around 500mV) and the integration controller reference voltage (around 1V for $\mathcal{R} = \{2, 2, \ldots, 2\}$). Additionally, the memory row 473 was set (storing the value 15) and the time stamp (`STAMP<3:0>` bus in Figure 4-17) was set to 0. These conditions simulate a pixel which has been reset in the previous check cycle and whose output is below the threshold at the moment of the check. With this scheme, the conditional and unconditional reset cycles take the same amount of time, $T_{UR} = T_{CR} = T_R$ and thus $T_{TS} = max\,(T_R, T_{ADC})$.

The quantization and data transmission operation proved to be the limiting factor in the light integration scheme. The chip has a mismatch between the parallelism of the integration controller and the ADC/CDS array. It takes 10 quantization operations to process an entire row while it only takes 1 check operation to process an entire row. After a quantization operation has finished, the outputs of the 64 ADCs have to be transmitted from the integrated circuit to the computer. While the computer data acquisition card specifications nominally allowed for transmission of a full resolution frame (10 bits, 640×480 pixels) in approximately 1msec, in practice the data rate had to be significantly reduced (by a factor of 8) to achieve error–free transmission. Unfortunately, the time shift in the rows also limits the minimum integration slot length, as $T_{TS} \equiv T_{MIN}$ where $T_{MIN} = min(\mathcal{T})$ since the next predictive pixel saturation check of a particular row can only occur after all the other rows have been checked.

The slow test board/computer data transmission rate would severely limit the maximum dynamic range increase E_{MAX}, so the shutter functionality of the pixel was used to remove the quantization and data transmission operations from the integration timing (Figure 5-2). However, this new timing configuration requires that the pixel voltage at the end of the integration time be held on the sensing node (node "SNS" of Figure 4-5) for a long period of time. Leakage current of the sensing node and potentially some photogenerated carriers diffusing from the photodiode alter the pixel voltage, lowering its value. Since lower pixel voltages translate into brighter gray levels once the CDS operation is performed, this effect "whitens" the frame from the upper left corner to the lower right corner (the order in which the rows are processed), as it can be seen on Figure 5-3.

A sequential integrating scheme was adopted to eliminate the light signal corruption (Figure 5-4). In this case, a 64–pixel region integrates light and its data is transmitted to the computer before the integration for the next 64–pixel region begins. The maximum
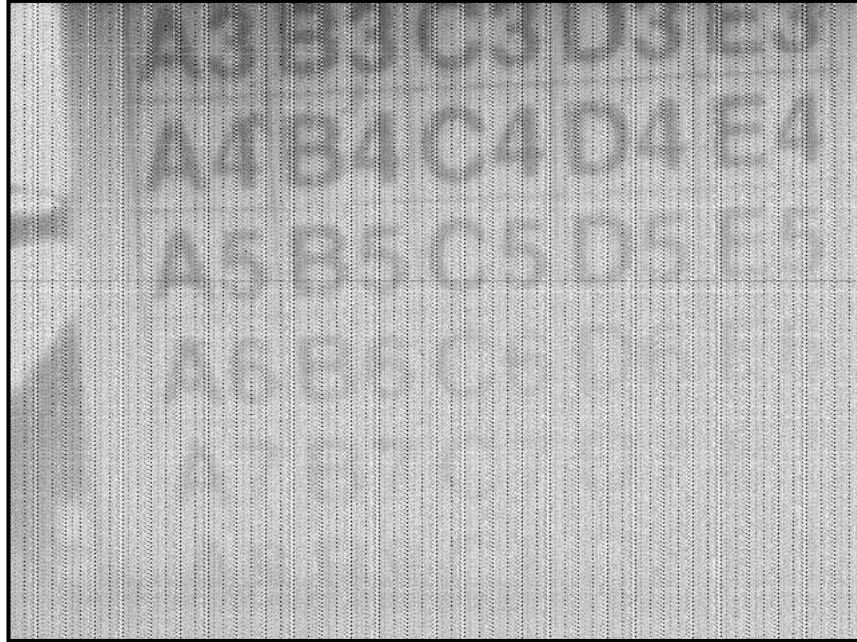
Figure 5-3: Image taken with the rolling shutter scheme using the shutter
functionality of the pixels. The "whitening" effect of the pro-
longed pixel voltage hold time in the sensing node can be seen
more markedly on the lower right corner. Rows read from upper
left corner to lower right corner.
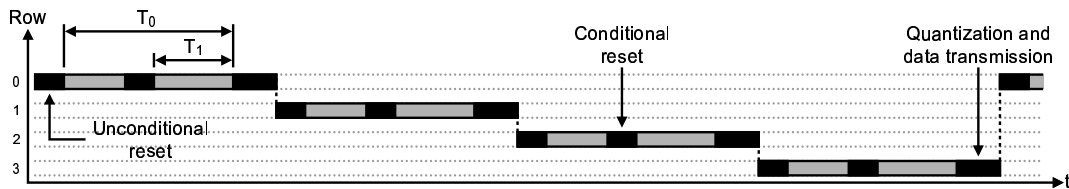
Figure 5-4: Light integration timing when the sequential scheme is used.
Illumination dynamic range expansion is maximized at the ex-
pense of frame rate.

dynamic range increase E_{MAX} in this scheme is substantially higher than in the rolling
shutter scheme since only one row is checked at a time. However, this increase comes at
the expense of a significantly reduced frame rate. A typical rolling shutter frame rate is
30frames/sec, while this sequential integrating scheme can only achieve 5frames/min with
the hardware available in the test bench.

Though the pixel shutter transistor $M1$ can be configured as a cascode amplifier to
increase the pixel sensitivity, this functionality was not used because that same sensitivity
makes the voltage at the pixel sensing node (node "SNS" in Figure 4-5) extremely suscepti-
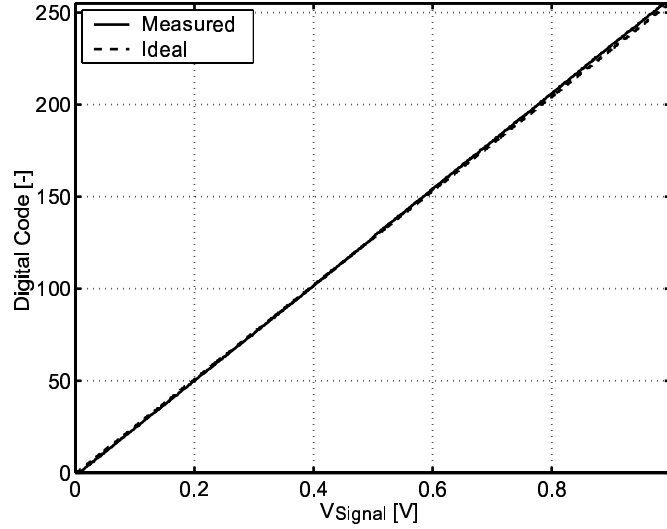
Figure 5-5: Measured ADC transfer characteristic for 8–bit resolution.

ble to pixel–to–pixel process parameter variations (leakage current, actual node capacitance, etc.). These variations introduce differences in the pixel output signals significantly increasing the pixel–to–pixel fixed pattern noise (FPN).

5.3   Analog–To–Digital Converter

Data from the proof–of–concept prototype was taken in digital format from the integrated ADC/CDS array. Therefore, the performance of this block affects all following measurements. To characterize the converters, a CDS operation followed by an 8–bit conversion was performed on externally–generated (single–ended) voltages varying from 1.5V to 0.5V with a reset voltage of 1.5V, thus approximating both the voltage swing and offset of the pixel output. The external voltage was input to the ADC/CDS array via the `CSVEQ` line (Figure 4-10), through the analog multiplexer.

The transfer characteristic of the ADC can be seen in Figure 5-5 (8–bit resolution). A linear fit to the measured data produced an offset of $D_{OS} \approx -1.89$digital codes, and a gain $A_{ADC} \approx 259.7551$digital codes/volt. These values make the measured transfer characteristic to be slightly below the ideal transfer characteristic for the $[0V, 0.3675V]$ input voltage interval, and slightly above for the remainder of the input voltage range.

Additionally, some variability around a mean converter output was observed when several samples of a constant input voltage (or illumination) were taken. Based on this em-
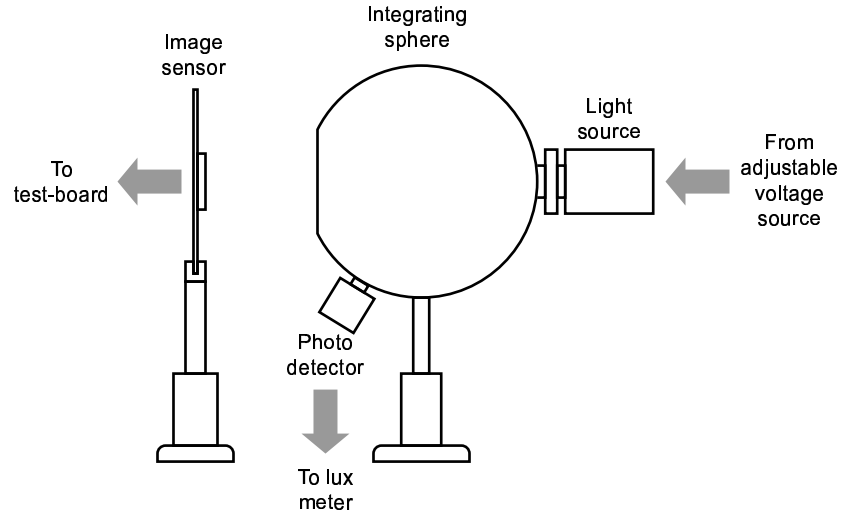
Figure 5-6: Test setup used to measure the image sensor transfer character-
istic and signal–to–noise ratio.

pirical data, the output of the ADCs was modeled as Gaussian (normal) variable with a standard deviation $\tau_{ADC} \approx 4$ from the sample mean corresponding to a particular input voltage (or illumination).

5.4   Transfer Characteristic

The transfer characteristic is one of the key parameters of the image sensor. The test setup shown in Figure 5-6 was used to obtain the required data. A 120W tungsten halogen light source with a color temperature of 3200°K is connected to the input port of an integrating sphere. This sphere has two output ports, one which is connected to the image sensor and one which is connected to a photosensor. Inside the sphere there is a baffle so that the output ports do not receive any direct light from the light source, and the inside of the sphere is specially coated so that it reflects all wavelengths equally. The net result is that the light coming out of the output ports appears to be coming from a point light source, i.e. the light power is constant for a given radius. The photodetector is connected to a lux meter to measure the light power received by the image sensor. The light source is connected to a adjustable voltage source so that different light power levels can be easily achieved.

For any illumination 100 frames were taken under identical measurement conditions.
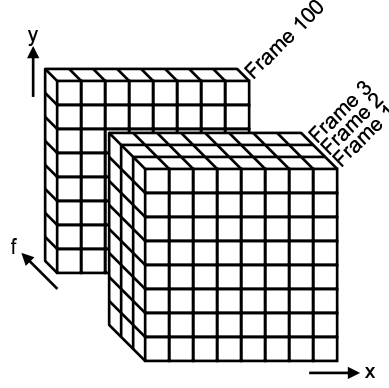
Figure 5-7: Visualization of the data set used to obtain the image sensor transfer characteristic and signal–to–noise ratio.

This data can be seen as a set $D(x, y, f, I, T_j)$ where x represent the image columns, y represent the image rows, f represents the frame number, I represents the illumination at which the frames were taken and T_j represents the integration slot used (Figure 5-7). To remove any constant offset that the analog signal pipeline might have, data was taken with no illumination for the longest integration slot used. Then the pixel offset O for every pixel was calculated as follows:

$$O(x, y, T_j) = \frac{1}{E_j} \cdot avg\left[D(x, y, f, 0, T_{INT})\right]_f \ \forall \ E_j \in \mathcal{E} \tag{5.2}$$

where $avg\left[\cdot\right]_f$ denotes the averaging operation over the f axis. The total signal for every pixel, for every frame was then zeroed with this offset and calculated as:

$$S_{TOT}(x, y, f, I) = E_i(x, y) \cdot (D(x, y, f, I, T_i) - O(x, y, T_i)) + A(i) \tag{5.3}$$

where the information of which integration slot each pixel uses was obtained from the memory contents $M(x, y)$ and $A(i)$ is the code shift that needs to be added to obtain a strictly linear transfer characteristic:

$$A(i) = \frac{E_i(x, y) - E_{i-1}(x, y)}{2} + \sum_{n=0}^{i-2} A(n), \text{ with } A(0) = 0 \tag{5.4}$$

Then the total pixel signal for every pixel was calculated:

$$S_{TOT}(x, y, I) = avg\left[S_{TOT}(x, y, f, I)\right]_f \tag{5.5}$$
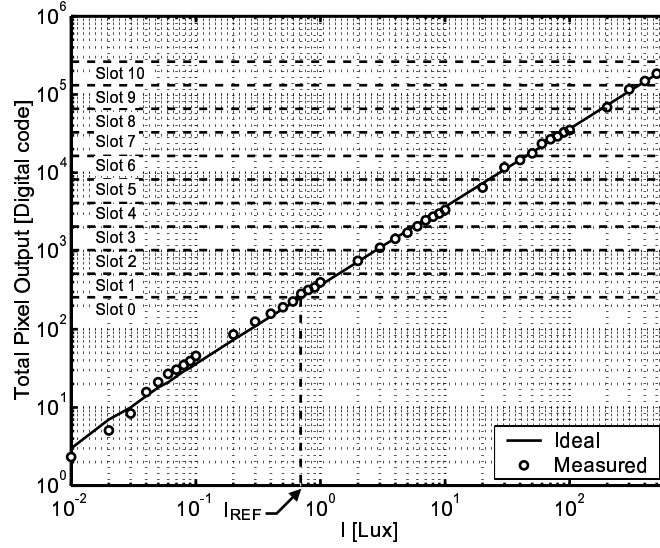
Figure 5-8: Measured   image   sensor   transfer   characteristic
with $T_{INT} \approx 30$msec   and $\mathcal{E} = \{2^z : z = 0, 1, \ldots 13\}$.
$I_{REF} = I_{TH}\,(T_{INT}) \approx 7 \cdot 10^{-1}$Lux.   Dashed   horizontal   lines
represent integration slot transitions.

To avoid introducing the mismatches between elements in the ADC array to the computations, the signal for the frame was calculated as follows:

$$S_{TOT}\,(I) = avg\,[S_{TOT}\,(x, y, I)]_{x \bmod 64, y \bmod 3} \tag{5.6}$$

where $avg\,[\cdot]_{x \bmod 64, y \bmod 3}$ denotes the averaging operation over the x and y axis taking into account that pixel outputs in column x are quantized by the same converter every 3 rows, and that pixel outputs in row y are quantized by the same converter every 64 columns. The illumination was swept from 10^{-2}Lux to $6 \cdot 10^{+2}$Lux with an integration time of $T_{INT} = T_0 = 30$msec and an integration slot set with 14 elements whose exposure ratio set is $\mathcal{E} = \{2^z : z = 0, 1, \ldots 13\}$. The results are shown in Figure 5-8 referenced by the ideal transfer characteristic as given by:

$$S_{TOT}\,(T_i) = E_i \cdot q\left(\frac{1}{E_i} \cdot \frac{I}{I_{REF}}\right) + A\,(i) \ , i \in \{0, 1, \ldots, M - 1\} \tag{5.7}$$

It can be seen that, as expected, the sensor responds linearly over 6 decades of illumination. Out of the 14 available integration slots, 11 are used for the measured illumination,

therefore the dynamic range increase provided by the algorithm is $1024\times$ or 60dB[1]. The reference illumination is $I_{REF} = I_{TH}(T_{INT}) \approx 7 \cdot 10^{-1}\text{Lux}$.

5.5 Responsivity

The responsivity measures the incremental change in the pixel output for an incremental change in illumination. Formally, the responsivity is the point–to–point derivative of the sensor transfer characteristic normalized by the integration time. A linear fit to the transfer characteristic data was performed and the results were:

$$S_{TOT_{Fit}}(I) \approx 369 \cdot I \tag{5.8}$$

while the linear interpolation of the ideal transfer characteristic has:

$$S_{TOT_{Ideal}}(I) = \frac{2^N}{I_{REF}} \cdot I \approx 366 \cdot I \ \ \text{for } N = 8\text{bits and } I_{REF} = 7 \cdot 10^{-1}\text{Lux} \tag{5.9}$$

Therefore for an integration time of $T_{INT} = 30\text{msec}$, the measured responsivity of the image sensor is[2]:

$$Resp \approx 48 \ \frac{\text{V}}{\text{Lux} \cdot \text{sec}} \tag{5.10}$$

The ideal responsivity is approximately 47.62 V/(Lux·sec).

5.6 Signal–to–noise Ratio

The same data that was used to obtain the transfer characteristic was used to obtain the SNR. In this case the variance of each pixel was calculated as follows:

$$N^2(x, y, I) = var\left[S_{TOT}(x, y, f, I)\right]_f \tag{5.11}$$

where $var\left[\cdot\right]_f$ denotes the variance calculation of the samples along the f axis. The noise of each pixel is assumed to be uncorrelated to the other pixels, so the total noise for the sensor is:

$$N(I) = \sqrt{avg\left[N^2(x, y, I)\right]_{x \bmod 64, y \bmod 3}} \tag{5.12}$$

---

[1]Maximum illumination limited by light source used during characterization.
[2]Assumes 1 digital code $\approx 4\text{mV}$
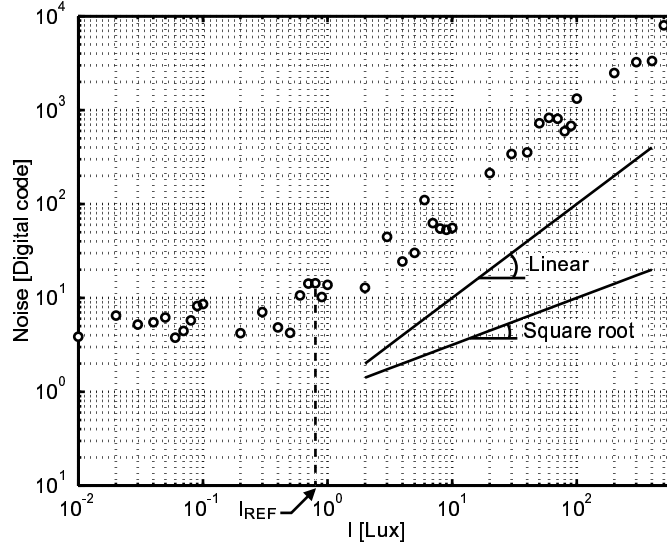
Figure 5-9: Measured   image   sensor   noise   with $T_{INT} \approx 30$msec   and $\mathcal{E} = \{2^z : z = 0, 1, \ldots 13\}$. $I_{REF} = I_{TH}(T_{INT}) \approx 7 \cdot 10^{-1}$Lux.

Figure 5-9 shows the noise contribution at every measured illumination level. The total noise remains flat for low illuminations and then increases for higher illuminations When photon shot noise overwhelms the noise from the analog readout circuitry the noise increases with a square root dependence (Equation 3.14). However, the measured noise increases with a linear dependence. Additionally, the average noise level for low illumination is approximately 4 (same as the empirically observed converter noise), and the start of the linear dependence also coincides with the transition between the longest integration slot and shorter ones ($I_{REF} \approx 7 \cdot 10^{-1}$Lux). Therefore, the measured noise at higher illuminations could simply be the ADC noise being scaled by the computation of the total pixel output required by the multiple sampling algorithm, and the SNR has to be observed to determine if this is the case. Formally, the signal–to–noise ratio was calculated as:

$$SNR(I) = \frac{S(I)}{N(I)} \tag{5.13}$$

Figure 5-10 shows the SNR for the measured illuminations, with $T_{INT} \approx 30$msec and $\mathcal{E} = \{2^z : z = 0, 1, \ldots 13\}$. It can be seen that the peak signal–to–noise ratio achieved close to the slot transitions is:

$$SNR_{MAX} \approx 35.593\text{dB} \tag{5.14}$$
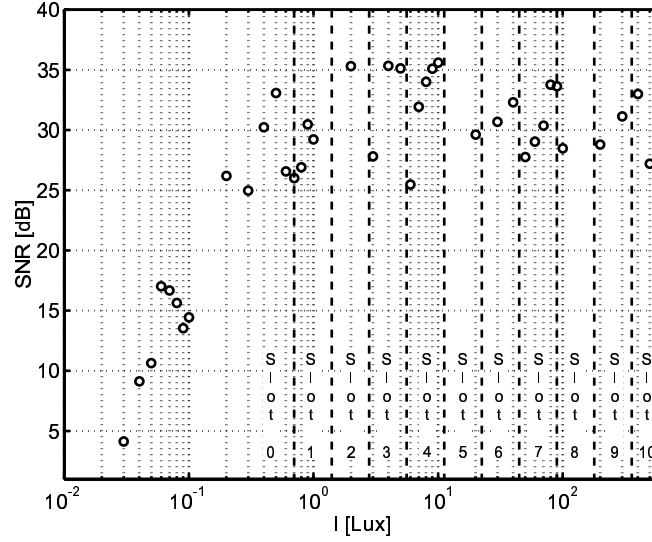
Figure 5-10: Measured image sensor signal–to–noise ratio with $T_{INT} \approx 30$msec and $\mathcal{E} = \{2^z : z = 0, 1, \ldots 13\}$. $I_{REF} = I_{TH}(T_{INT}) \approx 7 \cdot 10^{-1}$Lux. Dashed vertical lines indicate the threshold illuminations that bound the range corresponding to each integration slot.

Slightly before each slot transition the pixel is close to its maximum voltage swing and consequently the ADCs are close to full scale. Therefore, with an ADC standard deviation of 4 digital codes at an 8–bit level:

$$SNR_{ADC} = 20 \cdot log_{10}\left(\frac{2^N}{4}\right) = 20 \cdot log_{10}\left(\frac{256}{4}\right) = 36\text{dB} \approx SNR_{MAX} \qquad (5.15)$$

which is consistent with the hypothesis that the ADC noise dominates over all other noise sources. Even when this is the case, the characteristic "sawtooth" shape of the SNR (shown for the ideal case in Figure 5-10) should still be seen since for illuminations slightly bigger than the threshold illuminations $E_j \cdot I_{REF}, \forall E_j \in \mathcal{E}$ that just saturate the pixel for each integration slot:

$$SNR_{MIN} \approx 20 \cdot log_{10}\left(\frac{\frac{2^N}{R}}{4}\right) = 20 \cdot log_{10}\left(\frac{128}{4}\right) = 30\text{dB} \qquad (5.16)$$

using the fact that in the measurements $R_j = R = 2$ as $\mathcal{R} = \{2, 2, \ldots, 2\}$ (therefore for other integration slot ratios the SNR drop might be significantly bigger).
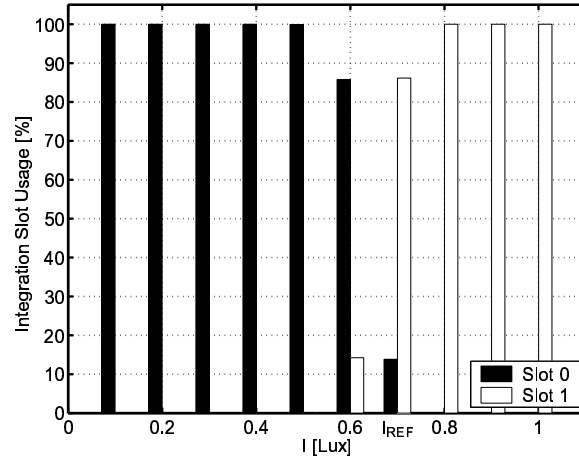
Figure 5-11: Integration slot usage in the $[0.1, 1]$Lux decade for 100 frames. Transition between integration slot 0 and integration slot 1 located approximately at $I_{REF} = 0.7$Lux. Pixels receiving an illumination well below the transition use integration slot 0 in all frames and pixels receiving an illumination well above the transition use integration slot 1 in all frames. However, pixels receiving an illumination close to the transition use integration slot 0 in some frames and integration slot 1 in other frames.
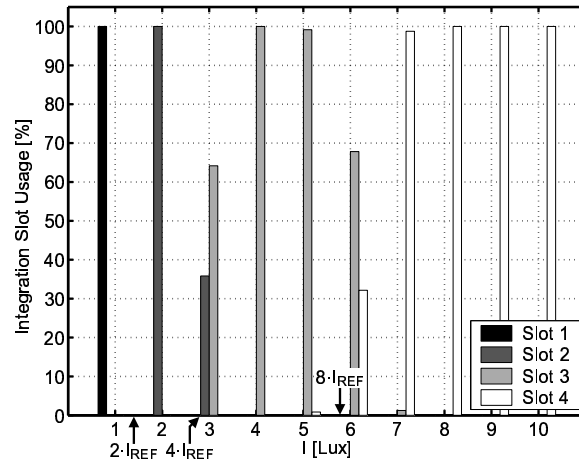
Figure 5-12: Integration slot usage in the $[1, 10]$Lux decade for 100 frames. Transition between integration slot 1 and integration slot 2 located approximately at 1.4Lux, transition between integration slot 2 and integration slot 3 located approximately at 2.8Lux, transition between integration slot 3 and integration slot 4 located approximately at 5.6Lux. Pixels receiving an illumination well below or above the transitions use a single integration slot in all frames, but pixels receiving an illumination close to the transitions use an integration slot for some frames and the next shortest or longest integration slot for other frames.

Within each integration slot the SNR should increase linearly from SNR_{MIN} to SNR_{MAX}, as it does in Figure 5-10. However, it can be observed that the minimum SNR is lower than the predicted 30dB, in fact for some transitions $SNR_{MAX} - SNR_{MIN} \approx 10$dB. This discrepancy occurs close to the slot transitions because for those illuminations pixels might not use a single integration slot for all frames.

Errors in the predictive saturation decision might lead pixels to use one integration slot for some frames and use an adjacent slot (next shorter or longer) for some other frames. This effect does not significantly alter the pixel signal but it does increase the noise (variance) of the sample. Figure 5-11 shows the integration slot usage for the $[0.1, 1]$Lux decade. Pixels that receive an illumination well below the transition I_{REF} use integration slot 0 in all frames and pixels receiving an illumination well above the transition use integration slot 1 in all frames. However, pixels receiving an illumination close to the transition use integration slot 0 in some frames and integration slot 1 in other frames. Figure 5-12 shows the integration slot usage for the $[1, 10]$Lux decade. Here it can also be seen that close to the transition points the slot usage is divided in non–negligible percentages between adjacent slots.

Sources of this unequal integration slot usage can be decomposed into two components: pixel–to–pixel and column–to–column. The main reason for column–to–column differences is the offset of the integration controller comparator. This is only present with column–parallel controllers, as in the case of the proof–of–concept IC, and can be eliminated from the data by only considering pixels of a single column. Pixel–to–pixel differences are those that occur in a single pixel column, i.e. those that occur even when the same integration controller is used. The pixel output V_{Pixel} during a predictive check is the output of the pixel source follower:

$$V_{Pixel}(t) = V_{O_{SF}} \left(V_{SNS_{RST}} - \frac{Q_{PH}(t)}{C_{ACC}} \right) \approx A_{SF} \cdot \left(V_{SNS_{RST}} - \frac{Q_{PH}(t)}{C_{ACC}} \right) - O_{SF} \quad (5.17)$$

where $V_{O_{SF}}$ is the output of a body–affected NMOS common drain amplifier (Section A.2), $V_{SNS_{RST}}$ is the reset voltage of the sensing node (Equation 4.3) and C_{ACC} is the capacitance where the photogenerated charges accumulate. $V_{O_{SF}}$ can be modeled with a gain term A_{SF} (neglecting a weak non–linear dependence on gate voltage) and an offset term O_{SF}. C_{ACC}

is:

$$C_{ACC} = \begin{cases} C_{PD} + C_{SNS} \approx C_{PD} & \text{, shutter transistor used as an always--on switch.} \\ C_{SNS} & \text{, shutter transistor used as cascode amplifier.} \end{cases}$$

(5.18)

with C_{PD} being the photodiode capacitance and C_{SNS} being the sensing node capacitance (in the proof--of--concept chip $C_{PD} \gg C_{SNS}$). If the input voltage range of the ADC is labeled ΔV_{ADC}, then the threshold voltage V_{REF_j} for the predictive check j is:

$$V_{REF_j} = V_{Pixel}(0) - \left(1 - \frac{1}{R_j}\right)\cdot\Delta V_{ADC} = A_{SF}\cdot V_{SNS_{RST}} - O_{SF} - \left(1 - \frac{1}{R_j}\right)\cdot\Delta V_{ADC} \quad (5.19)$$

where $Q_{PH}(0) = 0$ has been used. The pixel reset voltage $V_{SNS_{RST}}$ is different from frame to frame (even for a single pixel) due to kTC noise on the charge--accumulating node. The gate of pixel transistor $M1$ was left at a high potential to effectively make the photodiode the charge--accumulating node and therefore minimize the reset noise. The pixel reset voltage $V_{SNS_{RST}}$ can also be different from pixel to pixel due to its dependence on the parameters of the pixel transistors in the reset path, $M2$ and $M3$ (Section A.1). To eliminate this problem the high level of the RESPUL line was kept at a level such that $V_{SNS_{RST}} = V_{RESPUL_{HIGH}} = 2.5\text{V}$ (Equation 4.3). In other words, transistor $M3$ never enters the subthreshold regime[3]. The gain A_{SF} and offset O_{SF} of the source follower depend on transistor $M4$ parameters and geometry, so they are different from pixel to pixel and these differences cannot be offset.

For the integration slot set used to measure the SNR, the ideal integration slot ratio set is $R_j = R \; \forall \; j = 1, 2, \ldots, 12$ so $V_{REF_j} = V_{REF}$. However, from Equation 5.19:

$$R = \frac{\Delta V_{ADC}}{\Delta V_{ADC} + V_{REF} + O_{SF} - A_{SF} \cdot V_{SNS_{RST}}}$$

(5.20)

Consequently, any variation or noise in the reference voltage (it is an externally generated signal), pixel reset voltage, pixel output voltage or pixel--to--pixel variation in the source follower gain and/or offset effectively change the integration slot ratio, and thus the location of the transition between slots from frame to frame.

To verify that the uneven use of integration slots coupled with the ADC noise can indeed

---

[3]This is done at the expense of pixel signal swing.
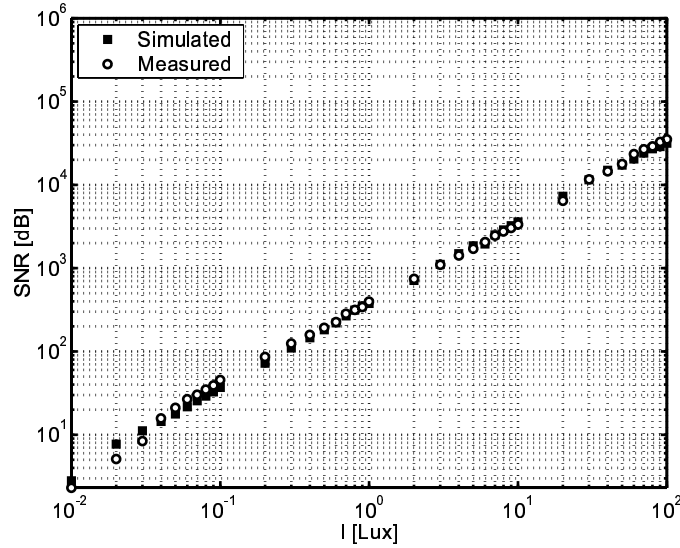
Figure 5-13: Comparison of measured transfer characteristic with simulated transfer characteristic that includes the effect of converter noise and errors in the predictive saturation decision.

lower SNR_{MIN} a Matlab simulation that implements the multiple sampling algorithm was run. In this simulation all system components were modeled ideally, except for the ADC and the predictive saturation decision. The ADC was modeled as a normal random variable with mean equal to the ideal quantized value given an input voltage and standard deviation of $\tau_{ADC} = 4$. A small random number was added to the sampled pixel voltage to simulate all possible sources of error in the predictive decision. The method used to obtain the simulated data mimics the method used to obtain the data from the proof–of–concept chip, 100 "frames" were taken and the results processed as outlined before.

Figure 5-13 shows the comparison between the measured and simulated transfer characteristic. Not surprisingly, since the mean of the ADC was taken as the output of an ideal quantizer, the two curves are nearly identical. Figure 5-14 shows the comparison between the measured and simulated image noise. A close match both in magnitude and in shape can be observed, which further confirms the hypothesis that the measured noise is scaled converter noise at higher illuminations. Finally, Figure 5-15 shows the comparison between the measured and simulated signal–to–noise ratio. The simulated values confirm the effects that the use of adjacent integration slots near the slot transitions have on SNR_{MIN}. The largest simulated SNR drop at the transitions is approximately 9dB.
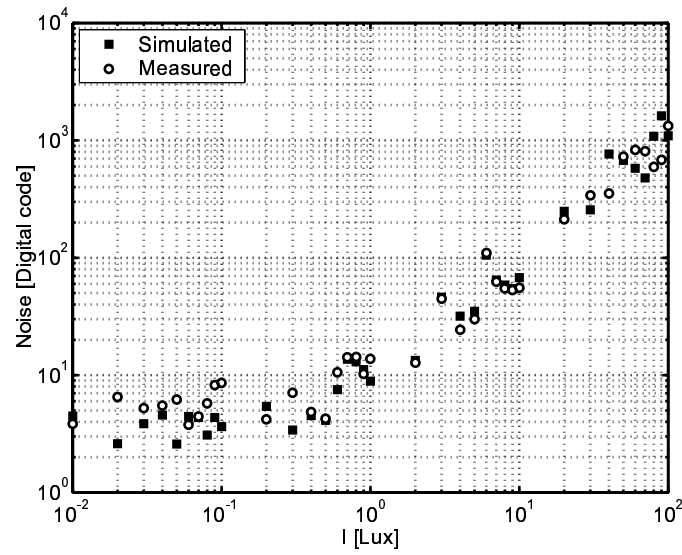
Figure 5-14: Comparison of measured noise with simulated noise that includes the effect of converter noise and errors in the predictive saturation decision.
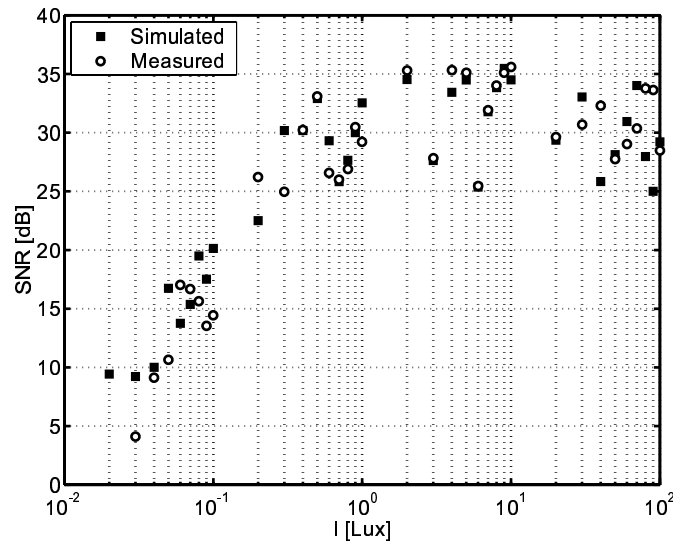
Figure 5-15: Comparison of measured signal–to–noise ratio with simulated signal–to–noise that includes the effect of converter noise and errors in the predictive saturation decision.
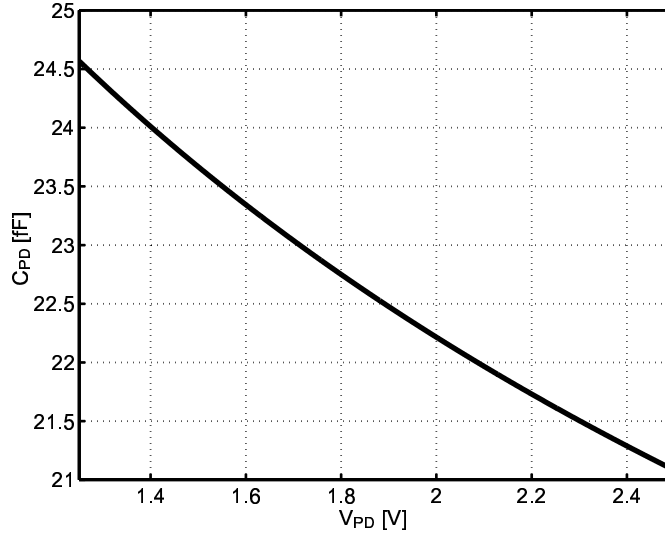
Figure 5-16: Capacitance variation for the allowed photodiode voltage swing.

5.7 Pixel Capacitance

The photodiode is an n^+–p substrate junction, therefore, its capacitance exhibits a non–linear behavior when reversed bias (the normal mode of operation in the pixel). A first order modeling of this effect is[4]:

$$C_{PD} \approx \frac{31.3\text{fF}}{\left(1 - \frac{V_{PD}}{0.8}\right)^{0.4}} + 2.4\text{fF} \tag{5.21}$$

where V_{PD} is the voltage at the photodiode node. The pixel output voltage swing is limited to 1V by the analog–to–digital converter, so with a measured pixel source follower gain of $A_{SF} \approx 0.8$ this allows a photodiode voltage swing of $\Delta V_{PD} \approx 1.25\text{V}$, from $V_{PD_{MAX}} = 2.5\text{V}$ (reset) to $V_{PD_{MIN}} = 1.25\text{V}$ (saturation). Figure 5-16 shows that for ΔV_{PD} the capacitance variation is in the [21.1fF, 24.6fF] range. The charge swing ΔQ_{PD} that results when a certain pixel receives an illumination I_{REF} can be calculated using Equation 5.21:

$$\Delta Q_{PD} = Q_{PD_{MAX}} - Q_{PD_{MIN}} = C_{PD}\left(V_{PD_{MAX}}\right) \cdot V_{PD_{MAX}} - C_{PD}\left(V_{PD_{MIN}}\right) \cdot V_{PD_{MIN}} \tag{5.22}$$

---

[4]Numerical values for the parameters are similar to those extracted from the fabrication process used.
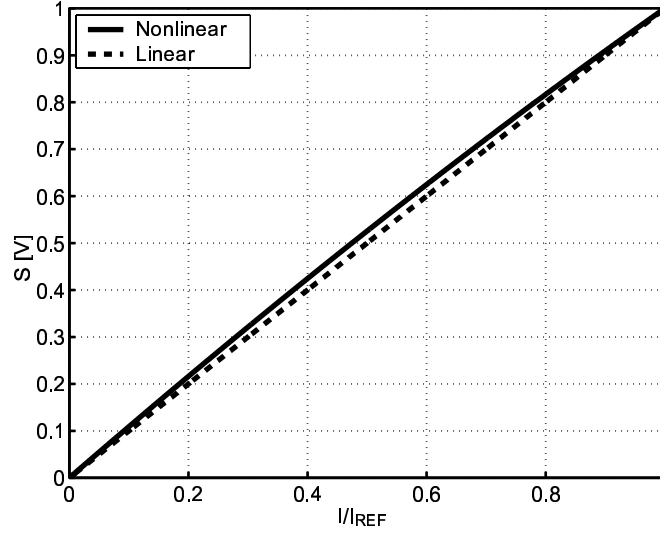
Figure 5-17: Pixel output voltage showing the effects of the non–linear pixel capacitance.

where $Q_{PD_{MAX}}$ corresponds to the reset level, and $Q_{PD_{MIN}}$ corresponds to the saturation level. The photodiode voltage can then be calculated for any accumulated photodiode charge $Q_{PD} \in [Q_{PD_{MIN}}, Q_{PD_{MAX}}]$ as the root of the following intrinsic equation:

$$Q_{PD} - C_{PD}(V_{PD}) \cdot V_{PD} = 0 \qquad (5.23)$$

 The pixel output signal (pixel output voltage with the reset level subtracted) calculated in this manner for the $[0, I_{REF}]$ illumination range can be seen in Figure 5-17. Not surprisingly, the photodiode capacitance non–linearity makes the pixel transfer characteristic also non–linear. For comparison purposes, a linear capacitance was calculated as:

$$C_{LIN} = \frac{\Delta Q_{PD}}{\Delta V_{PD}} \approx 17.6\text{fF} \qquad (5.24)$$

The pixel output difference can be defined as:

$$\Delta S\left(\frac{I}{I_{REF}}\right) = S_{NL}\left(\frac{I}{I_{REF}}\right) - S_L\left(\frac{I}{I_{REF}}\right) \qquad (5.25)$$

where $S_{NL}(I/I_{REF})$ denotes the non–linear pixel output and $S_L(I/I_{REF})$ denotes the linear pixel output for a given illumination I. From Figure 5-17, $\Delta S(I/I_{REF}) \geq 0$, $\Delta S(I/I_{REF}) = 0$ only when $I = 0$ or $I = I_{REF}$, and $\Delta S(I/I_{REF})_{MAX} \approx 25.6\text{mV}$ at $I = I_{REF}/2$.
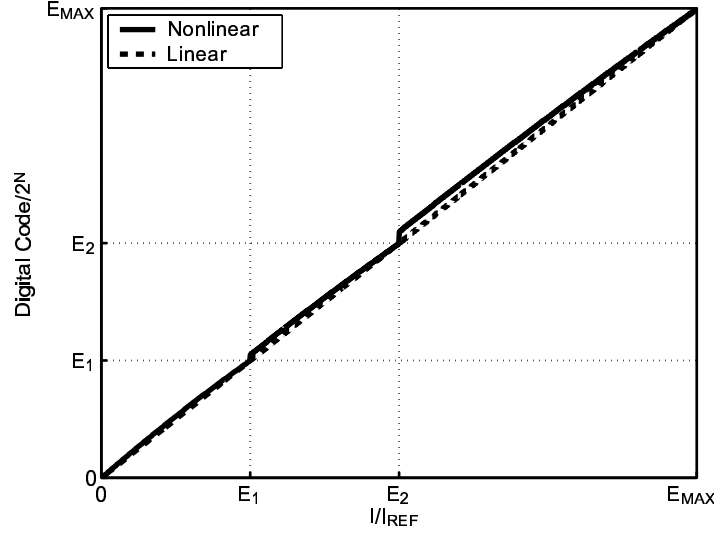
Figure 5-18: Sample transfer characteristic of an image sensor implementing the multiple sampling algorithm showing the effects of a non–linear pixel capacitance. $\mathcal{E} = \{1, 2, 4\}$ and $N = 8$ used.

To the first order the non–linearity of the pixel transfer characteristic is not a significant problem during the predictive pixel saturation checks because the reference voltage in the integration controller can be adjusted accordingly to account for this effect. However, inevitable pixel–to–pixel process variations in the photodiode capacitance will introduce some pixel–to–pixel errors in the total pixel output $S_{TOT}(T_i)$ (Equation 2.14) when the illumination received is close to one of the threshold illuminations $R_j \cdot I_{REF} \; \forall \; R_j \in \mathcal{R}$.

The photodiode capacitance non–linearity directly affects the quantized pixel output $S_q(T_i)$ so it will also affect the overall imager transfer characteristic as Figure 5-18 shows for a sensor having an exposure ratio set $\mathcal{E} = \{1, 2, 4\}$. Not only each section of the transfer characteristic is non–linear but there is also a pronounced increases in the digital code at the transition points. Since the non–linear total pixel output is:

$$S_{TOT}(T_i) = E_i \cdot q \left(\frac{1}{E_i} \cdot \frac{I}{I_{REF}} + \Delta S \left(\frac{I}{I_{REF}} \right) \right) + A(i) \tag{5.26}$$

then the digital code increase is on the order of $E_i \cdot q \left(\Delta S \left(I / I_{REF} \right) \right)$. The increases can be minimized when the integration ratios R_j are either very large or very close to unity since then $\Delta S \left(I / I_{REF} \right) \approx 0$, they are maximal when $R_j = 2$ as in the example of Figure 5-18.

While the non–linear effects can clearly be seen particularly for slot 0 in Figure 5-8, the

most important effect of the capacitance non–linearity is to contribute to the degradation of the SNR close to the transition points: when an error in the predictive decision occurs and a shorter integration slot is used, the resulting pixel output is bigger than ideal, the variance of the output (and thus the noise) increases and therefore the SNR decreases.

5.8   Sample Frames

A sample frame with localized high illumination was taken with the prototype image sensor using the same timing as in the transfer characteristic and SNR measurements. Figure 5-19 shows the original scene, with no dynamic range expansion. A toroidal light fixture with a magnifying glass in the middle partially blocks a target with labeled row and column cells. The light bulb and some areas of the target are completely saturated. Then the predictive checks were enabled and the raw pixel output ($S(T_i)$ in Equation 2.14) can be seen in Figure 5-20. Figure 5-21 shows the memory contents (used as indexes to a gray–scale color map, 0 for no pixel reset mapped to black), and Figure 5-22 shows the total pixel output. A fast bilateral filtering algorithm [70] was applied to the wide dynamic range image in order to adapt it to the lower dynamic range of the printer page without loosing significant details[5].

It can be seen from the memory contents that the center and upper left part of the scene saturate for T_0 while the lower left and upper right areas in the scene can still be properly imaged with this slot. The final image shows that the additional integration slots adaptively correct the exposure, to the point where details can be made out in the center, enough to improve the results of edge detection algorithms or other image processing algorithms.

5.9   Summary

The test setup to measure the performance of the proof–of–concept chip, the methodology used to obtain and process the data, and the experimental results have been presented. The sensor achieves significant ($1024\times$) linear dynamic range expansion when the sequential light integration scheme is used, with nearly theoretical responsivity. The signal–to–noise ratio is suboptimal due to the presence of high analog–to–digital converter noise. Simulations confirm that the measured drops in the SNR due to the multiple sampling algorithm,

---

[5]Processing software courtesy of Dr. F. Durand.
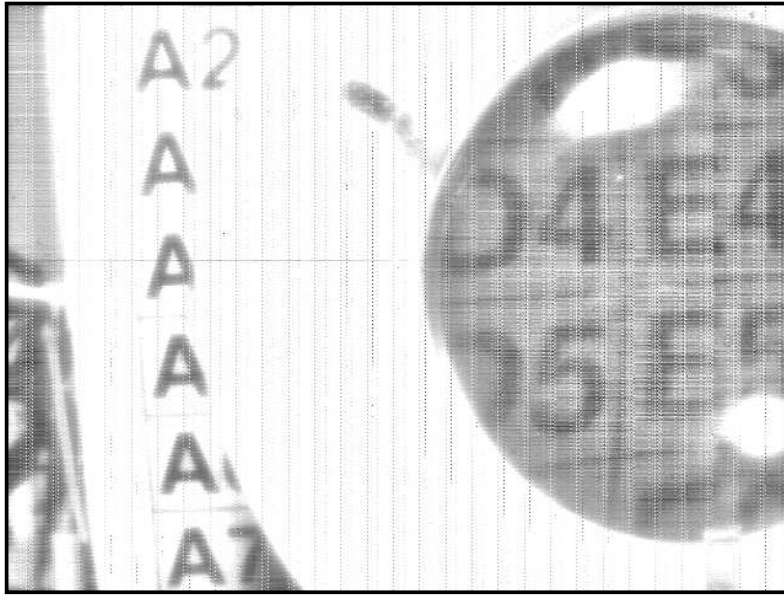
Figure 5-19: Sample image taken by the prototype image sensor with no wide dynamic range expansion. A toroidal light fixture with a magnifying glass in the middle partially blocks a target with labeled row and column cells. Light fixture and left side of the target are almost completely saturated.
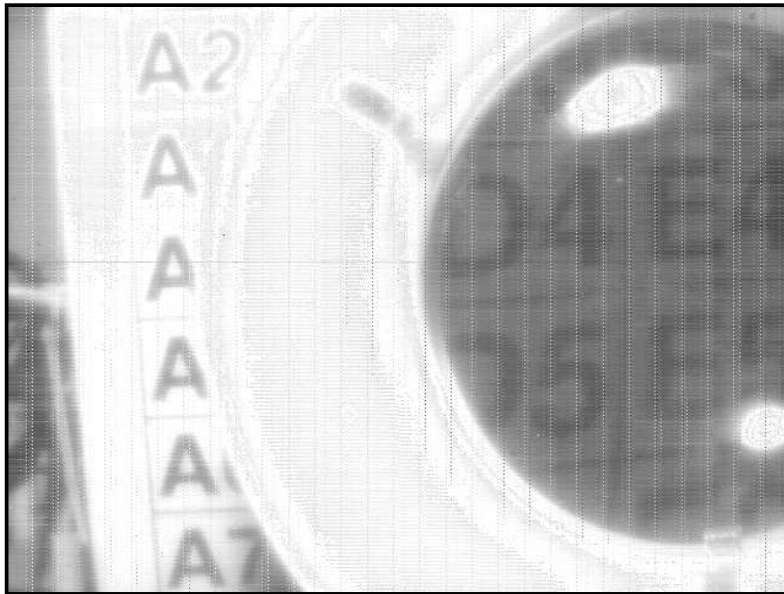
Figure 5-20: Quantized analog pixel output $S(T_i)$ with predictive checks enabled. The utilization of shorter integration slots can be detected as previously saturated areas now have values within the linear range. However, pixel scaling still remains to be done in order to obtain the wide dynamic range image.
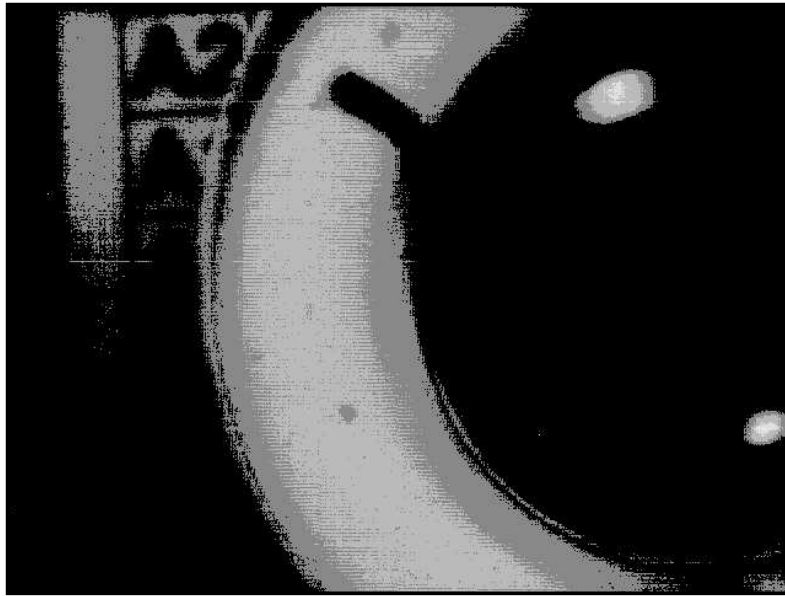
Figure 5-21: Memory contents when predictive checks are enabled used as indexes to a gray–scale color map. Black indicates the area where integration slot 0 was used, brighter colors indicate the areas where shorter integration slots were used.
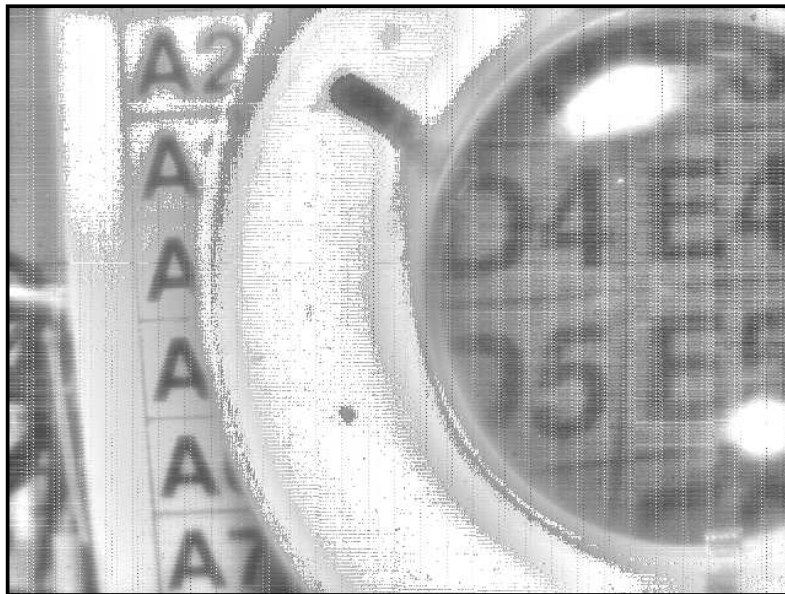
Figure 5-22: Total pixel output $S_{TOT}(T_i)$ obtained with the predictive checks enabled. The target grid on the left side can now be seen as well as details of the light fixture. The dark areas in the magnifying glass are still properly exposed.

Table 5.1
Proof–of–concept parameters and measured performance

Parameter	Value
Technology	0.18μm and 0.35μm devices 2–polysilicon layers, 5–metal layers p–well CMOS logic process
Total area	7600μm×11700μm
Power supplies	1.8V and 3.3V
Power dissipation	307mW
Sensing array format	1/3" VGA
Number of pixels	640(H)×480(V)
Pixel type	Photodiode
Pixel size	7.5μm×7.5μm
Fill factor	39% (Exposed photodiode area) 49% (Total photodiode area)
Photodiode capacitance	33.7fF @ 0V (Calculated)
Sensing node capacitance	1.7fF @ 0V (Calculated)
Pixel voltage range	1.25V
Pixel source follower gain	0.8
ADC input range	1V
ADC quantization noise	15.6mV
Responsivity	48 V/(Lux·sec)
Dynamic range expansion (sequential row processing)	> 60dB (1024×)

which were higher than anticipated, are affected by errors in the predictive saturation decision. Sample images confirm the correct and total functionality of the prototype.

Chapter 6

Conclusions

6.1 Summary

Wide illumination dynamic range is a desirable feature for high–end consumer applications, machine vision systems, and any other system that needs to extract information from the visible range of the electromagnetic spectrum. Reliably acquiring scenes with intensity differences exceeding $10^6 : 1$ is a challenge that has been faced using different techniques. The multiple sampling technique is an attractive option in this field because it can achieve the desired dynamic range increase linearly, preserving details at high illuminations and simplifying image post–processing. The main cost of the multiple sampling method is per–pixel memory, which is advisable to place in the same silicon die as the sensing array for performance reasons.

In the novel variant of the multiple sampling algorithm presented in this thesis, it is assumed that the illumination received by the pixels remains constant for a single frame, which then allows for integration periods (slots) of different duration to run concurrently by performing a predictive pixel saturation check at the potential start of every integration period. Two important contributions were made to the state–of–the–art of this method: a) a framework was developed to determine the optimal integration slot set composition given a certain set size for given illumination statistics, and b) the effects of the resulting sensor transfer characteristic on image data compression were delineated taking JPEG data compression as a case study.

A proof–of–concept integrated circuit was designed and fabricated. The chip integrates a VGA array, a 4bit per pixel SRAM, 64 cyclic ADC/CDS elements, and a column–parallel

integration controller that implements the main tasks required by the multiple sampling algorithm. Full functionality was observed, and illumination data shows linear dynamic range expansion with constant responsivity exceeding 60dB.

6.2   Future Work

Several improvements can be introduced to the existing proof–of–concept design to increase its functionality and efficiency:

- A way to unconditionally reset the pixels. This can be done by adding logic in the integration controller to bypass the predictive check decision and gate the `CPCOMP` line at will (Figure 4-17). Otherwise the cumbersome procedure outlined in Section 5.2 is needed, which both unnecessarily complicates the digital control and makes the unconditional reset operation longer than it needs to be.

- Implement the per–pixel storage using a dynamic random access memory (DRAM). This can lead to a smaller memory cell and therefore dramatically reduce the memory area. The requirement to periodically refresh the DRAM cells and the destructive nature of the DRAM read operation are two issues that are already addressed by the integration controller as currently implemented (Section 4.4). Every predictive check serves as a memory refresh cycle: the memory is read and after the saturation decision either the previously stored memory value is written (if the pixel is not reset) or the time stamp associated with the check is written (if the pixel is reset). For video frame rate (30frames/sec) the checks occur at worst every 33msec (if there is only one integration slot) which is on the same order of magnitude as typical DRAM refresh frequencies. A DRAM process option with high dielectric constant trench capacitors can also help provide significant reductions in the total memory area.

- The per–pixel memory size can further be optimized if the illumination statistics of the scene to be captured are known. In this case the optimal size and elements of the integration slot can be selected for a given I2D quantization noise (Sections 3.6 and 3.7), and the memory can then be sized to $\lceil \mathcal{T} \rceil$.

- If a fast read–out of region of interest (ROI) is not needed, the 2 additional output lines per pixel column can be removed. This increases the pixel fill factor (exposed

photodiode area in the particular case of the proof–of–concept IC pixel layout) a few percentage points, it reduces the area of the analog multiplexer, and it also eliminates the need to reorder the pixel stream. If row y is being read, the ADC array needs to be read following the order dictated by Equation 4.7 or otherwise the ADC array output stream needs to be rearranged according to this expression. The reordering operation adds complexity and post–processing time, furthermore the mod 3 operation involved in it might be difficult to implement in hardware (in the test setup it was done in software).

- Match the parallelism of the integration controller to the parallelism of the ADC array. The proof–of–concept chip had a column–parallel integration controller but only a 64 column–parallel ADC array. This forces the use of the shutter transistor to be able to integrate light on a full row basis. If image lag is a concern (there is extensive literature liking this effect to the use of an in–pixel electronic shutter transistor) the integration timing needs to be staggered not only one row with respect to another, but also one ADC column bank with respect to another within the same row, further complicating the digital control. Laying out a high–resolution ADC pitch–matched to the pixel width, which can certainly be smaller than the 7.5μm used in the prototype (for megapixel imagers), may prove to be a daunting task. Even if feasible the layout will certainly be extremely long and thus unattractive from a cost perspective. Therefore it is suggested that there should be as many integration controllers as ADCs can be reasonably fit in the sensing array width, provided all other performance requirements are satisfied.

The inaccuracies in the predictive saturation decision when the illumination is close to the thresholds highlight the need for an improved comparison scheme. If the area allotted for the integration controller allows it, offset–canceled comparators could be used thus also minimizing column–to–column mismatches. However, the comparison is still between a fixed reference voltage and the pixel signal whose starting point, the pixel reset voltage, is not known with certainty and varies from pixel to pixel. Ways to account for the pixel reset voltage would lead to more accurate, less noisy decisions and vastly better performance around the illumination thresholds.

Interesting possibilities can be realized when the high degree of programmability of the sensor is integrated and taken advantage of in a vision system. For example, the elements of the integration slot set, the region of interest to image, the frame rate, the dynamic range expansion and the quantization time can all be controlled and altered depending on the illumination received. Moreover, the memory contents are a low resolution, logarithmic–type image that can be accessed as a frame is being acquired. This not only provides early information that can be used to control mechanical systems or configure processing elements, but also, if the digital control is fast enough, it can help change sensor parameters on the fly, even before the integration time ends. The proof–of–concept integrated circuit is powerful and versatile hardware that can enhance and create exciting applications.

Bibliography

[1] N. Aleixos, J. Blasco, E. Moltó and F. Navarrón, "Assessment of Citrus Fruit Quality Using a Real–time Machine Vision System," in *Proc. 15th IEEE International Conference on Pattern Recognition*, vol. 1, Barcelona, Spain, Sept. 2000, pp. 482–485.

[2] J. Derganc and F. Pernus, "A Machine Vision System for Inspecting Bearings," in *Proc. 15th IEEE International Conference on Pattern Recognition*, vol. 4, Barcelona, Spain, Sept. 2000, pp. 752–755.

[3] K. Sobottka, and I. Pitas, "Extraction of Facial Regions and Features Using Color and Shape Information," in *Proc. 13th IEEE International Conference on Pattern Recognition*, vol. 3, Vienna, Austria, Aug. 1996, pp. 421–425.

[4] L. Li and M. Leung, "Fusion of Two Different Motion Cues for Intelligent Video Surveillance," in *Proc. IEEE Region 10 International Conference on Electrical and Electronic Technology*, vol. 1, Singapore, Aug. 2001, pp. 341–344.

[5] I. Masaki, "ASIC Approaches for Vision–Based Vehicle Guidance," *IEICE Transactions on Communications, Electronics, Information, and Systems*, vol. E76–C, no. 12, pp. 1735–1743, Dec. 1993.

[6] U. Franke, D. Gavrila, S. Görzig, F. Lindner F. Paetzold and C. Wöhler, "Autonomous Driving Approaches Downtown," *IEEE Intell. Syst.*, vol. 13, no. 6, pp. 40–48, Nov.–Dec. 1998.

[7] T. Naito, T. Tsukada, K. Yamada, K. Kozuka and S. Yamamoto, "Robust License–plate Recognition Method for Passing Vehicles under Outside Environment," *IEEE Trans. Veh. Technol.*, vol. 49, no. 6, pp. 2309–2319, Nov. 2000.

[8] Jeffrey C. Gealow, "An Integrated Computing Structure for Pixel–Parallel Image Processing," Ph.D. dissertation, Massachusetts Institute of Technology, Cambridge, Massachusetts, USA, June 1997.

[9] I. Masaki, "Machine–Vision Systems for Intelligent Transportation Systems," *IEEE Intell. Syst.*, vol. 13, no. 6, pp. 24–31, Nov.–Dec. 1998.

[10] J. Huang and D. Mumford, "Statistics of Natural Images and Models," in *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1, Fort Collins, CO, USA, June 1999, pp. 541–547.

[11] R. Dror, T. Leung, E. Andelson and A. Willsky, "Statistics of Real–world Illumination," in *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, Kauai, HI, USA, Dec. 2001, pp. 164–171.

[12] A. Theuwissen, *Solid-State Imaging with Charge-Coupled Devices.* Dordrecht, The Netherlands: Kluwer Academic Publishers, Mar. 1995.

[13] O. Yadid-Pecht, "Wide–Dynamic–Range Sensors," *Optical Engineering*, vol. 38, no. 10, pp. 1650–1660, Oct. 1999.

[14] R. Gonzalez and R. Woods, *Digital Image Processing.* Reading, Massachusetts: Addison–Wesley Publishing Company, 1993.

[15] A. El Gamal, "High Dynamic Range Image Sensors," in *IEEE International Solid State Circuit Conference Tutorial # 3*, San Francisco, CA, USA, Feb. 2002.

[16] S. Chamberlain and J. Lee, "A Novel Wide Dynamic Range Silicon Photodetector and Linear Imaging Array," *IEEE J. Solid-State Circuits*, vol. SC-19, no. 1, pp. 41–48, Feb. 1984.

[17] F. Boussaid, A. Bermak and A. Bouzerdoum, "A Wide Dynamic Range CMOS Imager with Extended Shunting Inhibition Image Processing Capabilities," in *Proc. IEEE International Symposium on Circuits and Systems*, vol. 2, Sydney, NSW, Australia, May 2001, pp. 620–623.

[18] B. Dierickx, D. Scheffer, G. Meynants, W. Ogiers and J. Vlummens, "Random Addressable Active Pixel Image Sensor," in *Proc. SPIE: Advanced Focal Plane Arrays and Electronic Cameras*, vol. 2950, Berlin, Germany, Oct. 1996, pp. 2–7.

[19] R. S. Muller and T. I. Kamins, *Device Electronics for Integrated Circuits*. New York, New York: John Wiley & Sons, 1996.

[20] C. Mead, *Analog VLSI and Neural Systems*. Reading, Massachusetts: Addison–Wesley Publishing Company, 1989.

[21] P. Gray and R. Meyer, *Analysis and Design of Analog Integrated Circuits*. New York, New York: John Wiley & Sons, 1993.

[22] Adaptive Photoreceptor with Wide Dynamic Range, "T. Delbrück and C. Mead," in *Proc. IEEE International Symposium on Circuits and Systems*, vol. 4, London, UK, May–June 1994, pp. 339–342.

[23] A. Andreou and K. Boahen, "A 590,000 Transistor 48,000 Pixel, Contrast Sensitive, Edge Enhancing, CMOS Imager –Silicon Retina–," in *Proc. 16th Conference on Advanced Research in VLSI*, Chapel Hill, NC, USA, Mar. 1995, pp. 225–240.

[24] K. Boahen, "A Retinomorphic Vision System," *IEEE Micro*, vol. 16, no. 5, pp. 30–39, Oct. 1996.

[25] M. Schwarz, R. Hauschild, B. Hosticka, J. Huppertz, T. Kneip, S. Kolnsberg, L. Ewe, and H. K. Trieu, "Single–chip CMOS Image Sensors for a Retina Implant System," *IEEE Trans. Circuits Syst. II*, vol. 46, no. 7, pp. 870–877, July 1999.

[26] M. Loose, K. Meier and J. Schemmel, "A Self–Calibrating Single–Chip CMOS Camera with Logarithmic Response," *IEEE J. Solid-State Circuits*, vol. 36, no. 4, pp. 586–596, Apr. 2001.

[27] V. Ward, M. Syrzycki, and G. Chapman, "CMOS Photodetector with Built–in Light Adaptation Mechanism," *Microelectronics Journal*, vol. 24, no. 5, pp. 547–554, Aug. 1993.

[28] M. Tabet and R. Hornsey, "CMOS Image Sensor Camera with Focal Plane Edge Detection," in *Canadian Conference on Electrical and Computer Engineering*, vol. 2, Toronto, ON, Canada, May 2001, pp. 1129–1134.

[29] A. Deval, A. Khan and L. Akers, "Locally Adaptive Active Pixel Sensors," in *Proc. IEEE International Symposium on Circuits and Systems*, vol. 1, Hong Kong, June 1997, pp. 685–688.

[30] F. Kub and G. Anderson, "Compressing Photodetectors for Long Optical Pulses Using a Lateral Blooming Drain Structure," *IEEE Trans. Electron Devices*, vol. 40, no. 10, pp. 1740–1744, Oct. 1993.

[31] T. Chan, O. Barrett, C. Chen, Y. Abendini and D. Wen, "A 488 × 430 Interline Transfer CCD Imager with Integrated Exposure and Blooming Control," in *Proc. IEEE International Solid–State Circuits Conference*, San Francisco, CA, USA, Feb. 1984, pp. 30–31.

[32] S. Decker, D. McGrath, K. Brehmer and C. Sodini, "A 256 × 256 CMOS Imaging Array with Wide Dynamic Range Pixels and Column–parallel Digital Output," *IEEE J. Solid-State Circuits*, vol. 33, no. 12, pp. 2081–2091, Dec. 1998.

[33] A Signal–Processing CMOS Image Sensor Using a Simple Analog Operation, "Y. Muramatsu, S. Kurosawa, M. Furumiya, H. Ohkubo and Y. Nakashiba," *IEEE J. Solid-State Circuits*, vol. 38, no. 1, pp. 101–106, Jan. 2003.

[34] O. Vietze and P. Seitz, "Image Sensing with Programmable Offset Pixels for Increased Dynamic Range of More Than 150dB," in *Proc. SPIE: Solid State Sensor Arrays and CCD Cameras*, vol. 2654, San Jose, CA, USA, Jan.–Feb. 1996, pp. 93–98.

[35] W. Yang, "A Wide–Dynamic–Range, Low–Power Photosensor Array," in *Proc. IEEE International Solid–State Circuits Conference*, San Francisco, CA, USA, Feb. 1994, pp. 230–231.

[36] L. McIlrath, "A Low–Power Low–Noise Ultra–Wide–Dynamic–Range CMOS Imager with Pixel–Parallel A/D Conversion," *IEEE J. Solid-State Circuits*, vol. 36, no. 5, pp. 846–853, May 2001.

[37] J. Rhee and Y. Joo, "Wide Dynamic Range CMOS Image Sensor with Pixel Level ADC," *Electronics Letters*, vol. 39, no. 4, pp. 360–361, Feb. 2003.

[38] B. Fowler, A. El Gamal and D. Yang, "A CMOS Area Image Sensor with Pixel–Level A/D Conversion," in *Proc. IEEE International Solid State Circuits Conference*, San Francisco, CA, USA, Feb. 1994, pp. 226–227.

[39] J. Raynor and P. Seitz, "A Linear Array of Photodetectors with Wide Dynamic Range and Near Photon Quantum–noise Limit," *Sensors and Actuators, A: Physical*, vol. 61, pp. 327–330, June 1997.

[40] O. Yadid–Pecht, R. Ginosar and Y. Diamand, "A Random Access Photodiode Array for Intelligent Image Capture," *IEEE Trans. Electron Devices*, vol. 38, no. 8, pp. 1772–1780, Aug. 1991.

[41] B. Cochrun and R. Taylor, "Linear Infrared Charge–Coupled Device (IRCCD) Focal Plane with Programmable Stare Time," in *Proc. SPIE: State–of–the–Art Imaging Arrays and their Applications*, vol. 501, San Diego, CA, USA, 1984, pp. 157–164.

[42] C. Huat and B. Wooley, "A 128×128–Pixel Standard–CMOS Image Sensor with Electronic Shutter," *IEEE J. Solid-State Circuits*, vol. 31, no. 12, pp. 1922–1930, Dec. 1996.

[43] R. Nixon, S. Kemeny, C. Staller and E. Fossum, "128×128 CMOS Photodiode–type Active Pixel Sensor with On–chip Timing, Control, and Signal Chain Electronics," in *Proc. SPIE: Charge–Coupled Devices and Solid State Optical Sensors V*, vol. 2415, San Jose, CA, USA, 1995, pp. 117–123.

[44] S. Chen and R. Ginosar, "An Adaptive Sensitivity^{TM} TDI CCD Sensor," in *Proc. SPIE: Advanced Focal Plane Arrays and Electronic Cameras*, vol. 2950, Berlin, Germany, Oct. 1996, pp. 45–48.

[45] ——, "An Adaptive Sensitivity^{TM} CCD Image Sensor," in *Proc. SPIE: Charge-Coupled Devices and Solid State Optical Sensors V*, vol. 2415, San Jose, CA, USA, Feb. 1995, pp. 303–309.

[46] O. Schrey, J. Huppertz, G. Filimonovic, A. Bußmann, W. Brockherde and B. Hosticka, "A 1K×1K High Dynamic Range CMOS Image Sensor with On–Chip Programmable

Region–of–interest Readout," *IEEE J. Solid-State Circuits*, vol. 37, no. 7, pp. 911–915, July 2002.

[47] O. Yadid–Pecht and E. Fossum, "Wide Intrascene Dynamic Range CMOS APS Using Dual Sampling," *IEEE Trans. Electron Devices*, vol. 44, no. 10, pp. 1721–1723, Oct. 1997.

[48] M. Schanz, C. Nitta, A. Bußmann, B. Hosticka and R. Wertheimer, "A High–Dynamic–Range CMOS Image Sensor for Automotive Applications," *IEEE J. Solid-State Circuits*, vol. 35, no. 7, pp. 932–938, July 2000.

[49] D. Yang, A. El Gamal, B. Fowler and H. Tian, "A 640×512 CMOS Image Sensor with Ultrawide Dynamic Range Floating–point Pixel–level ADC," *IEEE J. Solid-State Circuits*, vol. 34, no. 12, pp. 1821–1834, Dec. 1999.

[50] J.–L. Trépanier, M. Sawan, Y. Audet and K. Coulombe, "A New CMOS Architecture for Wide Dynamic Range Image Sensing," in *Proc. Canadian Conference on Electrical and Computer Engineering*, Montreal, Canada, May 2003, pp. 323–326.

[51] T. Lulé, M. Wagner, M. Verhoeven, H. Keller and M. Böhm, "100 000–Pixel, 120–dB Imager in TFA Technology," *IEEE J. Solid-State Circuits*, vol. 35, no. 5, pp. 732–739, May 2000.

[52] S.–H. Yang and K.–R. Cho, "High Dynamic Range CMOS Image Sensor with Conditional Reset," in *Proc. IEEE Custom Integrated Circuits Conference*, Orlando, FL, USA, May 2002, pp. 265–268.

[53] O. Yadid–Pecht and A. Belenky, "In–Pixel Autoexposure CMOS APS," *IEEE J. Solid-State Circuits*, vol. 38, no. 8, pp. 1425–1428, Aug. 2003.

[54] T. Yasuda, T. Hamamoto and K. Aizawa, "Adaptive–Integration–Time Image Sensor with Real–Time Reconstruction Function," *IEEE Trans. Electron Devices*, vol. 50, no. 1, pp. 111–120, Jan. 2003.

[55] S. Kleinfelder, S. Lim, X. Liu and A. El Gamal, "A 10,000 Frames/s CMOS Digital Pixel Sensor," *IEEE J. Solid-State Circuits*, vol. 36, no. 12, pp. 2049–2059, Dec. 2001.

[56] Y. Muramatsu, S. Kurosawa, M. Furumiya, H. Ohkubo and Y. Nakashiba, "A Signal–Processing CMOS Image Sensor using a Simple Analog Operation," in *Proc. IEEE International Solid State Circuits Conference*, San Francisco, CA, USA, Feb. 2001, pp. 98–99.

[57] D. Yang, A. El Gamal, B. Fowler and H. Tian, "A 640 × 512 CMOS Image Sensor with Ultra Wide Dynamic Range Floating–Point Pixel–Level ADC," in *Proc. IEEE International Solid State Circuits Conference*, San Francisco, CA, USA, Feb. 1999, pp. 308–309.

[58] P. Acosta–Serafini, I. Masaki and C. Sodini, "A Wide Dynamic Range Imager," U.S. patent application pending.

[59] G. Ward, "The RADIANCE Lighting Simulation and Rendering System," in *Proc. SIGGRAPH Conference*, Orlando, FL, USA, July 1994, pp. 459–472. [Online]. Available: radsite.lbl.gov/radiance

[60] "Matlab," The MathWorks, Inc., 3 Apple Hill Drive, Natick, MA 01760-2098, USA. [Online]. Available: www.mathworks.com

[61] K. Wallace, "The JPEG Still Picture Compression Standard," *Communications of the ACM*, vol. 34, no. 4, pp. 30–44, Apr. 1991.

[62] ——, "The JPEG Still Picture Compression Standard," *IEEE Trans. Consumer Electron.*, vol. 38, no. 1, pp. xviii–xxxiv, Feb. 1992.

[63] O. Yadid-Pecht; B. Pain; C. Staller; C. Clark and E. Fossum, "CMOS Active Pixel Sensor Star Tracker with Regional Electronic Shutter," *IEEE J. Solid-State Circuits*, vol. 32, no. 2, pp. 285–288, Feb. 1997.

[64] B. Pain, G. Yang, M. Ortiz, C. Wrigley, B. Hancock and T. Cunningham, "Analysis and Enhancement of Low–Light–Level Performance of Photodiode–type CMOS Active Pixel Imagers Operated With Sub–threshold Reset," in *IEEE Workshop Charge–Coupled Devices and Advanced Image Sensors*, Nagano, Japan, June 1999, pp. 140–143.

[65] R. Howe and C. Sodini, *Microelectronics: An Integrated Approach.* Upper Saddle River, New Jersey, USA: Prentice Hall, 1996, ch. 3, p. 912.

[66] K. Findlater, "A CMOS Camera Employing a Double Junction Active Pixel," Ph.D. dissertation, The University of Edinburgh, Edinburgh, Scotland, UK, June 2001.

[67] R. Chwang, M. Choi, D. Creek, S. Stern, P. Pelley, III, J. Schutz, P. Warkentin, M. Bohr and K. Yu, "A 70ns High Density 64K CMOS Dynamic RAM," *IEEE J. Solid-State Circuits*, vol. SC–18, no. 5, pp. 457–463, Oct. 1983.

[68] "LabVIEW," National Instruments Corp., 11500 N Mopac Expressway, Austin, TX 78759-35043, USA. [Online]. Available: www.ni.com

[69] Xilinx, Inc., San Jose, CA, USA. [Online]. Available: www.xilinx.com

[70] F. Durand and J. Dorsey, "Fast Bilateral Filtering for the Display of High-Dynamic-Range Images," *ACM Transactions on Graphics*, vol. 21, no. 3, pp. 257–266, July 2002.

Appendix A

Useful Mathematical Derivations

A.1 Maximum Source Terminal Voltage When Body–affected NMOS Transistor Used as a Switch to Charge a High Impedance Node

When the substrate (bulk) of an NMOS transistor is not connected to its source terminal, the threshold voltage (V_T) depends on the bulk–to–source voltage [65]:

$$V_T\left(V_{BS}\right) = V_{T_0} + \gamma \cdot \left(\sqrt{-2 \cdot \phi_P - V_{BS}} - \sqrt{-2 \cdot \phi_P} \right) \tag{A.1}$$

where V_{T_0} is the zero bias threshold voltage, γ is the body effect parameter and ϕ_P is the bulk potential when intrinsic silicon is taken as the reference. This source voltage dependence has a significant effect if the transistor is used a switch to charge a high impedance (capacitive) node (Figure A-1). Assuming that the gate of the transistor is at a voltage V_G and that the bulk is grounded ($V_B = 0$):

$$V_S = V_G - \left[V_{T_0} + \gamma \cdot \left(\sqrt{-2 \cdot \phi_P - (-V_S)} - \sqrt{-2 \cdot \phi_P} \right) \right] \tag{A.2}$$

This leads to a quadratic equation of the form:

$$a \cdot V_S^2 + b \cdot V_S + c = 0 \tag{A.3}$$

171
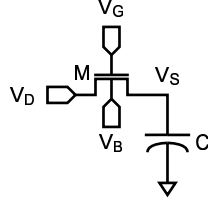
Figure A-1: NMOS transistor used a switch to charge a high impedance node.

with

$$a = \left(\frac{1}{\gamma}\right)^2 \tag{A.4}$$

$$b = \frac{2}{\gamma} \cdot \left(\frac{V_{T_0} - V_G}{\gamma} - \sqrt{-2 \cdot \phi_P}\right) - 1 \tag{A.5}$$

$$c = 2 \cdot \phi_P + \left(\frac{V_{T_0} - V_G}{\gamma} - \sqrt{-2 \cdot \phi_P}\right)^2 \tag{A.6}$$

After solving and rearranging terms:

$$
\begin{aligned}
V_S\left(V_G\right) = & V_G - V_{T_0} + \gamma \cdot \sqrt{-2 \cdot \phi_P} + \frac{\gamma^2}{2} - \\
& - \gamma \cdot \sqrt{V_G - V_{T_0} + \gamma \cdot \sqrt{-2 \cdot \phi_P} + \left(\frac{\gamma}{2}\right)^2 - 2 \cdot \phi_P}
\end{aligned}
\tag{A.7}
$$

Note that if $\gamma = 0$ (no body effect) the source potential reverts to the usual $V_G - V_{T_0}$.
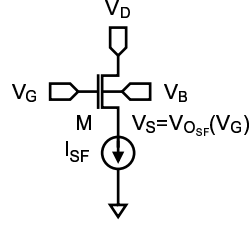
A.2   Voltage Offset of a Body–affected NMOS Common Drain Amplifier

The threshold voltage dependence on the bulk–to–source voltage also affect the input to output voltage shift of a source follower amplifier (Figure A-2). When the output has settled to its final value [65]:

$$I_{DS} = \frac{\mu_n \cdot C_{OX}}{2} \cdot \frac{W}{L} \cdot \left(V_{GS} - V_T\left(V_{BS}\right)\right)^2 = \frac{\mu_n \cdot C_{OX}}{2} \cdot \frac{W}{L} \cdot \left(V_G - V_S - V_T\left(-V_S\right)\right)^2 \tag{A.8}$$

which leads to:

$$V_S = V_G - \sqrt{\frac{2 \cdot I_{DS}}{\mu_n \cdot C_{OX} \cdot \frac{W}{L}}} - V_T\left(-V_S\right) \tag{A.9}$$

Figure A-2: Body–affected NMOS source follower.

and labeling

$$V_G' = V_G - \sqrt{\frac{2 \cdot I_{DS}}{\mu_n \cdot C_{OX} \cdot \frac{W}{L}}}$$ (A.10)

then

$$V_S = V_G' - \left[V_{T_0} + \gamma \cdot \left(\sqrt{-2 \cdot \phi_P - (-V_S)} - \sqrt{-2 \cdot \phi_P} \right) \right]$$ (A.11)

which is Equation A.2 when $V_G = V_G'$. Therefore the output voltage of the source follower $V_{O_{SF}}(V_G) = V_S(V_G)$ is:

$$V_{O_{SF}}(V_G) = V_G - \sqrt{\frac{2 \cdot I_{DS}}{\mu_n \cdot C_{OX} \cdot \frac{W}{L}}} - V_{T_0} + \gamma \cdot \sqrt{-2 \cdot \phi_P} + \frac{\gamma^2}{2} - \\ - \gamma \cdot \sqrt{V_G - \sqrt{\frac{2 \cdot I_{DS}}{\mu_n \cdot C_{OX} \cdot \frac{W}{L}}} - V_{T_0} + \gamma \cdot \sqrt{-2 \cdot \phi_P} + \left(\frac{\gamma}{2}\right)^2 - 2 \cdot \phi_P}$$ (A.12)