Post CMOS technologies: An Overview and Analysis

Willson Luo
willsonluo@ucla.edu
University of California Los Angeles
Los Angeles, California, USA

ABSTRACT

The rapid evolution of computing is pushing the limits of traditional silicon-based electronics, necessitating the exploration of novel materials and architectural paradigms. This paper investigates three transformative technologies poised to redefine future computational systems: Carbon Nanotubes (CNTs), Spintronics, and Memristors. We explore how Carbon Nanotube Field-Effect Transistors (CNFETs) offer enhanced performance and energy efficiency by enabling scaling beyond conventional limits, exemplified by the development of sophisticated CNFET-based microprocessors. Concurrently, Spintronics, particularly through Magnetoelectric Spin-Orbit (MESO) devices, promises ultra-low power logic by leveraging electron spin, introducing innovative concepts like majority logic gates. Furthermore, we examine Memristors, which address the Von Neumann bottleneck by integrating memory and processing, facilitating advanced applications in neuromorphic and analog computing, as demonstrated by their use in artificial neural networks and threshold logic gates. While each technology presents unique challenges in fabrication and integration, their distinct advantages offer compelling pathways toward building faster, smaller, and more energy-efficient computing systems for the next generation. CMOS.

KEYWORDS

Do, Not, Us, This, Code, Put, the, Correct, Terms, for, Your, Paper

ACM Reference Format:
Willson Luo. 2018. Post CMOS technologies: An Overview and Analysis. In *Proceedings of Make sure to enter the correct conference title from your rights confirmation email (Conference acronym 'XX)*. ACM, New York, NY, USA, 8 pages. https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

The pursuit of faster, smaller, and more energy-efficient computing has been a driving force for technological advancement since the creation of the transistor. For decades, CMOS (Complementary metal-oxide-semiconductor) technology has served as the foundation of of the modern computer. However, as we continue to approve our transistors and approach fundamental limits, the need for novel paradigms becomes increasinly urgent. This paper explores three technologies that could shape the future of computing: Carbon Nanotubes (CNTs), Spintronics, and Memristors. Each offers their own advantages that address the limitations of silicon transistors, varying from faster performance to significantly reduced power consumption.

Carbon Nanotubes, with their exceptional electrical and thermal properties, present a compelling alternative to silicon in field-effect transistors (FETs). The ability to scale devices beyond current limits, coupled with high carrier mobility and superior heat dissipation, positions Carbon Nanotube FETs (CNFETs) as a potential successor for high-performance, low-power logic. Recent advancements, such as the development of the RV16X-NANO microprocessor at MIT, demonstrate the practical viability of CNFET-based computing, showcasing their compatibility with existing design methodologies and their ability to execute complex instructions.

Beyond simply replacing silicon, Spintronics offers a revolutionary approach to information processing by leveraging the intrinsic spin of electrons in addition to their charge. Magnetoelectric Spin-Orbit (MESO) devices, a key development in spintronics, promise ultra-low power logic operations by directly coupling electron spin with its movement and enabling control through both electrical and magnetic properties. These devices offer superior switching energy, lower operating voltages, and enhanced logic density, with the potential to implement universal logic through majority gates, fundamentally transforming how logic functions are conceived and executed.

Finally, Memristors, often described as the "fourth fundamental circuit element," introduce a new dimension to computing by combining memory and processing capabilities within a single device. This integration directly addresses the Von Neumann bottleneck, the traditional separation of memory and logic that limits computational speed and efficiency. Memristor-based threshold logic gates and crossbar arrays hold immense promise for neuromorphic and analog computing, enabling highly efficient Vector-Matrix Multiplication (VMM) crucial for artificial neural networks. While challenges in manufacturing reliability and programmability persist, the ability of memristors to store multiple bits and their non-volatile nature offer a path toward more compact, power-efficient, and brain-inspired computing architectures.

This paper will delve into the principles, advancements, and challenges associated with each of these transformative technologies, highlighting their individual contributions and their collective potential to shape the next generation of computing systems.

2 CARBON NANOTUBES (CNTS) FOR ADVANCED ELECTRONICS

As the semiconductor continues its pursuit of miniaturization and greater performance, the fundamental limits of silicon-based transistors become increasingly apparent. This fundamental limit to the scaling of traditional transistors has given rise to many alternatives among them being Carbon Nanotubes (CNTs), one dimensional nanostructures that possess a unique combination of characteristics making them highly attractive for advancing electronics.

CNTs are composed of graphene sheets rolled into cylindrical tubes with their structure determining whether they behave as semiconductors or metals. his tunable electrical property, coupled with their nanoscale dimensions, high carrier mobility, and exceptional thermal conductivity, positions CNTs as a compelling material for high-performance computing. Their inherent flexibility also opens avenues for novel applications beyond traditional rigid electronics. This section will delve into the fundamental properties of CNTs, explore their application in Carbon Nanotube Field-Effect Transistors (CNFETs), highlight recent advancements that demonstrate their potential, and discuss the significant fabrication challenges that must be overcome for their widespread adoption.

2.1 Fundamental Properties of Carbon Nanotubes

Carbon nanotubes are made of an allotrope of carbon known as graphene.

Their structure can be characterized into to major types. Single-walled and multi-walled carbon nanotubes. Single-walled carbon nanotubes consist of a single graphene sheet rolled into a cylinder typically with a diameter between 0.5 to 2.0 nanometers. Multi-walled carbon nanotubes are comprised of multiple concentric graphene cylinders nested within each other. Their diameter can extend up to tens of nanometers.

The electrical properties can also vary depending on the structure and chirality (the angle at which the graphene sheet is rolled up) of the CNTs. This means that a CNT can behave either as metals with a high conductivity or semiconductors making them suitable for making transistors.

CNTs also have extremely high thermal conductivity making them suitable to overcome many of the power limitations of current CMOS scaling.

Finally, CNTs are among the strongest and stiffest materials known, possessing exceptional tensile strength and elastic modulus. Moreover, they are remarkably flexible, capable of withstanding considerable mechanical strain without fracturing. This flexibility is particularly advantageous for emerging flexible electronics and 3D integration.

2.2 Carbon Nanotube Field Effect Transistors (CNFETs)

The ability of semiconductor CNTs to conduct extremely effectively and their nanoscale dimensions make them ideal candidates for making the channel in field effect transistors. These properties allow CNFETs to offer several significant advantages over silicon based FETs:

- Aggressive Scaling: Their nanoscale dimensions allow for device scaling below the 5 nanometer limit.
- High Carrier Mobility: The ballistic or near-ballistic transport of carriers within CNTs leads to very high carrier mobility, which translates directly to faster switching speeds and improved device performance.
- Lower Power Consumption: CNFETs can operate effectively at lower voltages due to their superior electrical characteristics, resulting in significantly reduced power dissipation.
- Enhanced Gate Control: As one-dimensional materials, CNTs offer excellent electrostatic control by the gate, leading to lower off-state currents and steeper subthreshold swings.
- Superior Thermal Management: Their high thermal conductivity facilitates efficient heat dissipation, mitigating hot spots and improving reliability in densely packed circuits.
- Material Flexibility: The mechanical flexibility of CNTs enables the development of bendable electronics and facilitates potential three-dimensional chip stacking and integration, offering new avenues for compact and innovative designs.

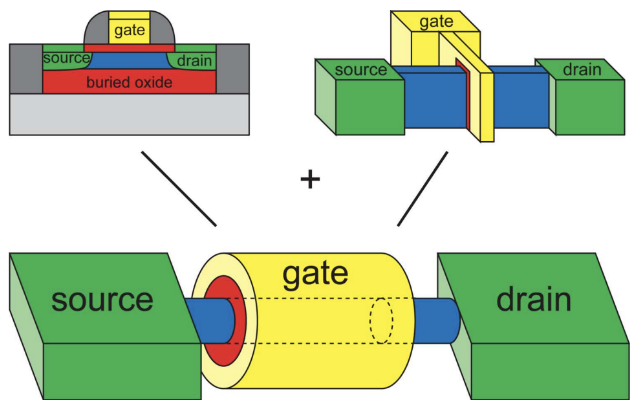

Figure 1: Diagram of CNFET as combination of MOSFET and FinFET

2.3 Logic Operations

Since CNFETs operate just like normal transistors their integration into creating logic gates is the exact same as a traditional transistor. They still follow CMOS architecture and their benefits come from their performance advantage compared to traditional transistors rather than new operating paradigms.

2.4 Recent Advancements: RV16X-NANO Microprocessor

The RV16X-NANO Microprocessor has been one of the biggest advancements in making a computer with CNFETs. The MIT Medical Electronic Device Realization Center were able to develop a 16 bit nanoprocessor which they named RV16X-NANO. The microprocessor was built of the RISC-V instruction set which was written through Bluespec and then compiled into Verilog, an RTL hardware description langauge. The microprocessor was comprised of over 14000 CMOS CNFETs that were made using more than 10 million CNTs.

On this microprocessor they were able to implement many standard logic blocks like multiplexers, arithmetic logic units, decoders, and encoders. Most notably, they were able to execute the famous "Hello World" program.

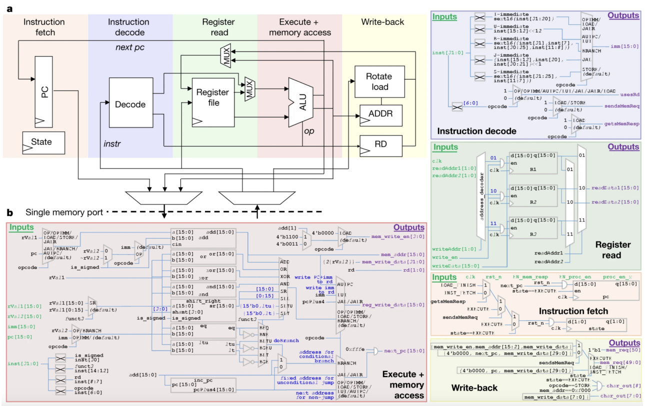

Figure 2: Architecture and Design of RV16X-NANO

2.5 Challenges in CNFET Fabrication and Integration

Despite the considerable amount of progress made, several challenges still need to be addressed before CNFETs can achieve widespread viability and meet yield requirements for very large scale integration (VLSI). Most signficantly, material and manufacturing defects are the current biggest challenge. Current methods of CNT fabrication lack a reliable way of controlling the chirality leading to many CNTs exhibiting metallic properties instead of being semiconductors. Even after achieving semiconducting CNTs, manufacturing processes still introduce defects and variability across a wafer. This includes variation in CNFET density and contact resistance leading to non uniform performance across an entire chip. In addition to manufacturing issues there still lacks a way of seamlessly integrating with traditional silicon CMOS components. Overcoming these challenges is crucial to achieving

2.6 Personal Thoughts

Overall CNFETs seem like a very promising area where there could be a lot of development. Since it doesn't deviate signficantly from traditional transistors it can be implemented into already made computing schemes making it more appealing for companies to invest in. It is much more likely for top fabrication companies like TSMC or Intel to look into CNFETs than some other technologies that present completely new computing paradigms. The progress made in CNFETs has also been extremely promising as demonstrated by MIT's RV16X-NANO computer. The milestone to already have demonstrated such funcionality proves that this is a technology that is really feasible for the future. Graphene is also an area that has a good amount of research going into it so carbon nanotubes in general are sure to get more attention.

With new chips there is a greater emphasis on chiplet technology as well as 3D integration. Again, CNFETs look very promising in that area. Along with that, the medical industry is also constantly looking for flexible devices that can better interact with the human body which is another area where CNFETs shine in.

Though there are still some big hurdles to overcome in manufacturing CNFETs at large scales and reliably I believe with just a few more breakthroughs they can become quite mainstream where large amounts of investments will be poured into CNFETs and accelerate their development greatly.

3 SPINTRONICS: BEYOND CHARGE-BASED COMPUTING

Traditional electronics primarily rely on the charge of electrons to process and store information. 1s and 0s are coded by high and low volages which are determined by electric fields created by distributions of charges. However, this approach faces inherent limitations in terms of power consumption and switching speed as devices continue to shrink. Spintronics emerges as a new paradigm that seeks to overcome these limitations by exploiting not only electron's charge but also its intrinsic angular momentum, known as spin. This additional degree of freedom offers the potential for fundamentally new device functionalities, leading to ultra-low power consumption, higher operating speeds, and increased logic density.

3.1 Introduction to Spintronics

Spintronics is the study and exploitation of the spin of electrons in solid-state devices. Unlike charge, which is a scalar quantity, spin is a quantum mechanical property that can be oriented in different directions (e.g., "spin-up" or "spin-down"). This allows for the encoding of information in a non-volatile magnetic state. The promise of spintronics lies in its ability to enable novel functionalities and overcome the inherent energy dissipation associated with moving charges in conventional electronics. By manipulating spin currents and magnetic states, spintronic devices aim to perform logic operations with significantly reduced energy footprints. By also utilizing using a charge as well as spin this gives the potential to encode more states where there can be a combination of high or low voltage along with up or down spin.

3.2 Magnetoelectric Spin-Orbit (MESO) Devices

One promising way to compute and perform logic using spintronics is Magnetoelectric Spin-Orbit (MESO). In this scheme binary is represented in the up or down spin state of a nanomagnet. These devices rely on 2 major physical effects. The first is spin-orbit transduction where if a current flows through a material with a strongly coupled spin the current will also gain that spin. The second effect is magnetoelectric switching where you can control a materials magnetic properties by applying an electric field. For instance switching the spin state with the help of an electric field.

3.2.1 Principle of Operation. An input logic is provided through a voltage and electric field which will set the magnetization in a magnet to be in a certain direction. This converts an electrical input into a magnetic state where the spin state can now be used to represent the binary 0 or 1.

To then generate an output a current is injected into the nanomagnet which will cause the output current from the nanomagnet to all have the same

4 MEMRISTORS

The Von Neumann architecture, which physically separates the processing unit from the memory has been the preferred method for making computers. However, as our chips become faster bottlenecks from I/O also come up where information can't be moved to and from the chip fast enough. Memristors, first described as the fourth missing circuit element, offer a way to link memory and compute in a single device. This combination opens up potential for more efficient brain inspired computing, particularly in the neuromorphic and analog world.

4.1 The Memristor Concept

A memristor, combination of the words memory and resistor, is a device that can have a programmble resistance. It was often described as the fourth circuit element because it was believed that magnetic fields could be used to change the resistance of these devices. However, recent devices don't rely on magnetic fields at all and instead use electric fields to change the physical and chemical properties of these devices to program the resistance. This property, where it can "remember" its resistance allows for non volatile memory and reconfigurable logic without having to change the physical connections on a chip.

In addition to these new computing regimes they also offer advantages in several other areas. They can be manufactured into nanoscale dimensions below those of a silicon transistor allowing for greater scaling and higher density integration. Their ability to store memory even when power is switched enables lower power consumption. Since they can also be programmed into continuous resistance values they have the ability to store more than just 1 bit per device. Recent experiments have shown one memristor being able to store 5-7 bits. Advances in memristors have also demonstrated a high level of CMOS compatability allowing them to be used in hybrid structures with traditional CMOS devices.

4.2 Memristor Based Threshold Logic Gate (TLG)

Beyond their abilities to store memory, memristors are perfectly suited for implementing threshold logic gates.

4.2.1 Threshold Logic Gate. A threshold logic gate is a gate that outputs a high signal if an input signal surpasses a certain threshold and outputs low otherwise. To give an example simply, imagine the input had a range of voltages from 0-9 and there is a threshold voltage of 5. Any input voltage above high will output a binary high signal from the gate and any voltage below will output a low signal. A TLG is oftentimes also used to analyze a weighted sum from its inputs and see if that is above the threshold.

With these characteristics a TLG can implement any linearly separable function. This means that for a given function, there exists a dividing line (or hyperplane in higher dimensions) that would separate all input combinations to one of two distinct output groups.

This also means that a single TLG can implement a boolean function that would require multiple logic gates to realize. For example, a majority gate would require at least 4 NAND gates to be realized in traditional CMOS design while it can be implemented in just one TLG.

4.2.2 Memristor-Based Threshold Logic Gate. Memristors are ideal for TLG implementation since their variable resistance can act as programmable weights. A common architecture for a memristor based TLG implements a hybrid architecture with memristors and traditional CMOS components into two main parts: a differential part and a sensor part.

The differential part is where a memristor is actually used. Each memristor is paired with a transistor known as a 1T1M array. Here an input and reference threshold signal is taken in. When a voltage is then applied as the input current flows through the memristor. The memristor here acts as a knob that controls how much current is passed through this differential part. This part can also be switched off when not in use to save power.

The current from the differential part then goes to the sensor. The sensor compares a combined current coming from either one or multiple differential parts and sees if its above the threshold. Based on this comparison either a 1 or 0 is outputted indicating whether the sum of the weights have surpassed the threshold.

4.2.3 Challenges. Despite the theoretical efficiency of memristor based TLGs, their practical implementation still face significant challenges. Fabrication is still very difficult and there are no EDA tools compatible with designing devices made of memristors TLGs.

4.3 Hardware Implementation of Memristor-Based Artificial Neural Networks

Beyond logic gates, memristors are particularly compelling for neural networks due to their inherent ability to perform efficient vector matrix multiplicaton which serves as the backbone of modern neural networks.

4.3.1 Vector-Matrix Multiplication (VMM) Core. The central component in the vector-matrix multiplication core is the memristor crossbar array. In this array, memrisotrs are placed at the crosspoints of a grid of wires. Voltages are inputted through the rows of the wire and the at each cross point the current is found through Ohm's Law. Kirchoff's current law then sums up all the currents through the vertical beams and the sum of these currents is the result of the vector multiplication. Mathematically the product as a current is

$$I_j = \Sigma_i V_i G_{ij}.$$

These raw analog values are then fed to a analog to digital converter (ADC) to then turn into digital magnitudes. Finally, an activation function (often a sigmoid or ReLU) is applied to these results to introduce nonlinearities, making the output resemble various desired distributions.

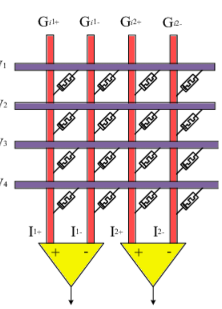

Figure 3: Memristor crossbar array

4.3.2 *Demonstrations of Memristor Neural Networks.*

Notably, the University of Michigan created a memristor based computer that utilized a 54x108 memristor crossbar array. With this computer they were able to create a single layer perception that classified a 5x5 array of pixels, identifying the greek letters drawn in the 25 pixel array with 100% accuracy. In addition, they implemented a two layer neural network that found commonalities and differences in breast cancer screenings, correctly classifying the cases as benign or malignant with 94.6% accuracy.

4.4 Challenges to Implementing Memristors

Despite their demonstrated potential, several critical challenges still prevent the widespread and practical use of memristors. Among these challenges the most prominent ones are the lack of reliable manufacturing ability, resistance drift, and non-linear resistance programming.

Memristors are still unable to be manufactured reliably and the the cost of metal alloys is much more than that of silicon. Manufacturing metals to these scales is also much harder. These manufactured devices also suffer from drift where the programmed resistance can change over time to deviations in the environment like temperature. Non-linearities in how the resistance is programmed also make it extremely difficult to program large scale memristor arrays. Supposed identitical memristors could require different voltage thresholds or to be applied for varying amount of times to program them to have the same resistance.

4.5 Personal Thoughts

With the advent of

5 COMPARATIVE ANALYSIS

6 CONCLUSION

7 ACKNOWLEDGMENTS

Authors of any work published by ACM will need to complete a rights form. Depending on the kind of work, and the rights management choice made by the author, this may be copyright transfer, permission, license, or an OA (open access) agreement.

Regardless of the rights management choice, the author will receive a copy of the completed rights form once it has been submitted. This form contains LaTeX commands that must be copied into the source document. When the document source is compiled, these commands and their parameters add formatted text to several areas of the final document:

- the "ACM Reference Format" text on the first page.
- the "rights management" text on the first page.
- the conference information in the page header(s).

Rights information is unique to the work; if you are preparing several works for an event, make sure to use the correct set of commands with each of the works.

The ACM Reference Format text is required for all articles over one page in length, and is optional for one-page articles (abstracts).

8 CCS CONCEPTS AND USER-DEFINED KEYWORDS

Two elements of the "acmart" document class provide powerful taxonomic tools for you to help readers find your work in an online search.

The ACM Computing Classification System — https://www.acm.org/publications/class-2012 — is a set of classifiers and concepts that describe the computing discipline. Authors can select entries from this classification system, via https://dl.acm.org/ccs/ccs.cfm, and generate the commands to be included in the LaTeX source.

User-defined keywords are a comma-separated list of words and phrases of the authors' choosing, providing a more flexible way of describing the research being presented.

CCS concepts and user-defined keywords are required for for all articles over two pages in length, and are optional for one- and two-page articles (or abstracts).

9 SECTIONING COMMANDS

Your work should use standard LaTeX sectioning commands: `\section`, `\subsection`, `\subsubsection`, `\paragraph`, and `\subparagraph`. The sectioning levels up to `\subsusection` should be numbered; do not remove the numbering from the commands.

Simulating a sectioning command by setting the first word or words of a paragraph in boldface or italicized text is **not allowed.**

Below are examples of sectioning commands.

9.1 Subsection

This is a subsection.

9.1.1 *Subsubsection.*

This is a subsubsection.

Paragraph. This is a paragraph.

Subparagraph This is a subparagraph.

10 TABLES

The "`acmart`" document class includes the "`booktabs`" package — https://ctan.org/pkg/booktabs — for preparing high-quality tables.

Table captions are placed *above* the table.

Because tables cannot be split across pages, the best placement for them is typically the top of the page nearest their initial cite. To ensure this proper "floating" placement of tables, use the environment **table** to enclose the table's contents and the table caption. The contents of the table itself must go in the **tabular** environment, to be aligned properly in rows and columns, with the desired horizontal and vertical rules. Again, detailed instructions on **tabular** material are found in the *LaTeX User's Guide*.

Immediately following this sentence is the point at which Table 1 is included in the input file; compare the placement of the table here with the table in the printed output of this document.

To set a wider table, which takes up the whole width of the page's live area, use the environment **table*** to enclose the table's contents and the table caption. As with a single-column table, this wide table will "float" to a location deemed more desirable. Immediately following this sentence is the point at which Table 2 is included in the input file; again, it is instructive to compare the placement of the table here with the table in the printed output of this document.

Table 1: Frequency of Special Characters

Non-English or Math	Frequency	Comments
Ø	1 in 1,000	For Swedish names
π	1 in 5	Common in math
\$	4 in 5	Used in business
Ψ_1^2	1 in 40,000	Unexplained usage

Always use midrule to separate table header rows from data rows, and use it only for this purpose. This enables assistive technologies to recognise table headers and support their users in navigating tables more easily.

11 MATH EQUATIONS

You may want to display math equations in three distinct styles: inline, numbered or non-numbered display. Each of the three are discussed in the next sections.

11.1 Inline (In-text) Equations

A formula that appears in the running text is called an inline or in-text formula. It is produced by the **math** environment, which can be invoked with the usual `\begin . . . \end` construction or with the short form `$ . . . $`. You can use any of the symbols and structures, from α to ω, available in LaTeX [24]; this section will simply show a few examples of in-text equations in context. Notice how this equation: $\lim_{n\rightarrow \infty}x=0$, set here in in-line math style, looks slightly different when set in display style. (See next section).

11.2 Display Equations

A numbered display equation—one set off by vertical space from the text and centered horizontally—is produced by the **equation** environment. An unnumbered display equation is produced by the **displaymath** environment.

Again, in either environment, you can use any of the symbols and structures available in LaTeX; this section will just give a couple of examples of display equations in context. First, consider the equation, shown as an inline equation above:

$$\lim_{n\rightarrow \infty}x=0 \tag{1}$$

Notice how it is formatted somewhat differently in the **displaymath** environment. Now, we'll enter an unnumbered equation:

$$\sum_{i=0}^{\infty} x + 1$$

and follow it with another numbered equation:

$$\sum_{i=0}^{\infty}x_i=\int_{0}^{\pi+2} f \tag{2}$$

just to demonstrate LaTeX's able handling of numbering.

12 FIGURES

The "`figure`" environment should be used for figures. One or more images can be placed within a figure. If your figure contains third-party material, you must clearly identify it as such, as shown in the example below.

Figure 4: 1907 Franklin Model D roadster. Photograph by Harris & Ewing, Inc. [Public domain], via Wikimedia Commons. (https://goo.gl/VLCRBB).

Your figures should contain a caption which describes the figure to the reader.

Figure captions are placed *below* the figure.

Every figure should also have a figure description unless it is purely decorative. These descriptions convey what's in the image to someone who cannot see it. They are also used by search engine crawlers for indexing images, and when images cannot be loaded.

A figure description must be unformatted plain text less than 2000 characters long (including spaces). **Figure descriptions should not repeat the figure caption – their purpose is to capture important information that is not already provided in the caption or the main text of the paper.** For figures that convey important and complex new information, a short text description may not be adequate. More complex alternative descriptions can be placed in an appendix and referenced in a short figure description. For example, provide a data table capturing the information in a bar chart, or a structured list representing a graph. For additional information regarding how best to write figure descriptions and why doing this is so important, please see https://www.acm.org/publications/taps/describing-figures/.

12.1 The "Teaser Figure"

A "teaser figure" is an image, or set of images in one figure, that are placed after all author and affiliation information, and before the body of the article, spanning the page. If you wish to have such a figure in your article, place the command immediately before the `\maketitle` command:

```
\begin{teaserfigure}
  \includegraphics[width=\textwidth]{sampleteaser}
  \caption{figure caption}
  \Description{figure description}
\end{teaserfigure}
```

13 CITATIONS AND BIBLIOGRAPHIES

The use of BibTeX for the preparation and formatting of one's references is strongly recommended. Authors' names should be complete — use full first names ("Donald E. Knuth") not initials ("D. E. Knuth") — and the salient identifying features of a reference should be included: title, year, volume, number, pages, article DOI, etc.

The bibliography is included in your source document with these two commands, placed just before the `\end{document}` command:

```
\bibliographystyle{ACM-Reference-Format}
\bibliography{bibfile}
```

Table 2: Some Typical Commands

Command	A Number	Comments
`\author`	100	Author
`\table`	300	For tables
`\table*`	400	For wider tables

where "`bibfile`" is the name, without the "`.bib`" suffix, of the BibTeX file.

Citations and references are numbered by default. A small number of ACM publications have citations and references formatted in the "author year" style; for these exceptions, please include this command in the **preamble** (before the command "`\begin{document}`") of your LaTeX source:

```
\citestyle{acmauthoryear}
```

Some examples. A paginated journal article [2], an enumerated journal article [10], a reference to an entire issue [9], a monograph (whole book) [23], a monograph/whole book in a series (see 2a in spec. document) [17], a divisible-book such as an anthology or compilation [12] followed by the same example, however we only output the series if the volume number is given [13] (so Editor00a's series should NOT be present since it has no vol. no.), a chapter in a divisible book [35], a chapter in a divisible book in a series [11], a multi-volume work as book [22], a couple of articles in a proceedings (of a conference, symposium, workshop for example) (paginated proceedings article) [3, 15], a proceedings article with all possible elements [34], an example of an enumerated proceedings article [14], an informally published work [16], a couple of preprints [6, 7], a doctoral dissertation [8], a master's thesis: [4], an online document / world wide web resource [1, 28, 36], a video game (Case 1) [27] and (Case 2) [26] and [25] and (Case 3) a patent [33], work accepted for publication [30], 'YYYYb'-test for prolific author [31] and [32]. Other cites might contain 'duplicate' DOI and URLs (some SIAM articles) [21]. Boris / Barbara Beeton: multi-volume works as books [19] and [18]. A couple of citations with DOIs: [20, 21]. Online citations: [36–38]. Artifacts: [29] and [5].

14 ACKNOWLEDGMENTS

Identification of funding sources and other support, and thanks to individuals and groups that assisted in the research and the preparation of the work should be included in an acknowledgment section, which is placed just before the reference section in your document.

This section has a special environment:

```
\begin{acks}
...
\end{acks}
```

so that the information contained therein can be more easily collected during the article metadata extraction phase, and to ensure consistency in the spelling of the section heading.

Authors should not prepare this section as a numbered or unnumbered `\section`; please use the "`acks`" environment.

15 APPENDICES

If your work needs an appendix, add it before the "`\end{document}`" command at the conclusion of your source document.

Start the appendix with the "`appendix`" command:

```
\appendix
```

and note that in the appendix, sections are lettered, not numbered. This document has two appendices, demonstrating the section and subsection identification method.

16 MULTI-LANGUAGE PAPERS

Papers may be written in languages other than English or include titles, subtitles, keywords and abstracts in different languages (as a rule, a paper in a language other than English should include an English title and an English abstract). Use `language=...` for every language used in the paper. The last language indicated is the main language of the paper. For example, a French paper with additional titles and abstracts in English and German may start with the following command

```
\documentclass[sigconf, language=english, language=german,
               language=french]{acmart}
```

The title, subtitle, keywords and abstract will be typeset in the main language of the paper. The commands `\translatedXXX`, `XXX` begin title, subtitle and keywords, can be used to set these elements in the other languages. The environment `translatedabstract` is used to set the translation of the abstract. These commands and environment have a mandatory first argument: the language of the second argument. See `sample-sigconf-i13n.tex` file for examples of their usage.

17 SIGCHI EXTENDED ABSTRACTS

The "`sigchi-a`" template style (available only in LaTeX and not in Word) produces a landscape-orientation formatted article, with a wide left margin. Three environments are available for use with the "`sigchi-a`" template style, and produce formatted output in the margin:

`sidebar`: Place formatted text in the margin.
`marginfigure`: Place a figure in the margin.
`margintable`: Place a table in the margin.

ACKNOWLEDGMENTS

To Robert, for the bagels and explaining CMYK and color spaces.

REFERENCES

[1] Rafal Ablamowicz and Bertfried Fauser. 2007. *CLIFFORD: a Maple 11 Package for Clifford Algebra Computations, version 11.* Retrieved February 28, 2008 from http://math.tntech.edu/rafal/cliff11/index.html
[2] Patricia S. Abril and Robert Plant. 2007. The patent holder's dilemma: Buy, sell, or troll? *Commun. ACM* 50, 1 (Jan. 2007), 36–44. doi:10.1145/1188913.1188915

[3] Sten Andler. 1979. Predicate Path expressions. In *Proceedings of the 6th. ACM SIGACT-SIGPLAN symposium on Principles of Programming Languages (POPL '79)*. ACM Press, New York, NY, 226–236. doi:10.1145/567752.567774

[4] David A. Anisi. 2003. *Optimal Motion Control of a Ground Vehicle*. Master's thesis. Royal Institute of Technology (KTH), Stockholm, Sweden.

[5] Sam Anzaroot and Andrew McCallum. 2013. *UMass Citation Field Extraction Dataset*. Retrieved May 27, 2019 from http://www.iesl.cs.umass.edu/data/data-umasscitationfield

[6] Sam Anzaroot, Alexandre Passos, David Belanger, and Andrew McCallum. 2014. Learning Soft Linear Constraints with Application to Citation Field Extraction. arXiv:1403.1349

[7] Lutz Bornmann, K. Brad Wray, and Robin Haunschild. 2019. Citation concept analysis (CCA)—A new form of citation analysis revealing the usefulness of concepts for other researchers illustrated by two exemplary case studies including classic books by Thomas S. Kuhn and Karl R. Popper. arXiv:1905.12410 [cs.DL]

[8] Kenneth L. Clarkson. 1985. *Algorithms for Closest-Point Problems (Computational Geometry)*. Ph. D. Dissertation. Stanford University, Palo Alto, CA. UMI Order Number: AAT 8506171.

[9] Jacques Cohen (Ed.). 1996. Special issue: Digital Libraries. *Commun. ACM* 39, 11 (Nov. 1996).

[10] Sarah Cohen, Werner Nutt, and Yehoshua Sagic. 2007. Deciding equivalances among conjunctive aggregate queries. *J. ACM* 54, 2, Article 5 (April 2007), 50 pages. doi:10.1145/1219092.1219093

[11] Bruce P. Douglass, David Harel, and Mark B. Trakhtenbrot. 1998. Statecarts in use: structured analysis and object-orientation. In *Lectures on Embedded Systems*, Grzegorz Rozenberg and Frits W. Vaandrager (Eds.). Lecture Notes in Computer Science, Vol. 1494. Springer-Verlag, London, 368–394. doi:10.1007/3-540-65193-4_29

[12] Ian Editor (Ed.). 2007. *The title of book one* (1st. ed.). The name of the series one, Vol. 9. University of Chicago Press, Chicago. doi:10.1007/3-540-09237-4

[13] Ian Editor (Ed.). 2008. *The title of book two* (2nd. ed.). University of Chicago Press, Chicago, Chapter 100. doi:10.1007/3-540-09237-4

[14] Matthew Van Gundy, Davide Balzarotti, and Giovanni Vigna. 2007. Catch me, if you can: Evading network signatures with web-based polymorphic worms. In *Proceedings of the first USENIX workshop on Offensive Technologies (WOOT '07)*. USENIX Association, Berkley, CA, Article 7, 9 pages.

[15] Torben Hagerup, Kurt Mehlhorn, and J. Ian Munro. 1993. Maintaining Discrete Probability Distributions Optimally. In *Proceedings of the 20th International Colloquium on Automata, Languages and Programming (Lecture Notes in Computer Science, Vol. 700)*. Springer-Verlag, Berlin, 253–264.

[16] David Harel. 1978. *LOGICS of Programs: AXIOMATICS and DESCRIPTIVE POWER*. MIT Research Lab Technical Report TR-200. Massachusetts Institute of Technology, Cambridge, MA.

[17] David Harel. 1979. *First-Order Dynamic Logic*. Lecture Notes in Computer Science, Vol. 68. Springer-Verlag, New York, NY. doi:10.1007/3-540-09237-4

[18] Lars Hörmander. 1985. *The analysis of linear partial differential operators. III*. Grundlehren der Mathematischen Wissenschaften [Fundamental Principles of Mathematical Sciences], Vol. 275. Springer-Verlag, Berlin, Germany. viii+525 pages. Pseudodifferential operators.

[19] Lars Hörmander. 1985. *The analysis of linear partial differential operators. IV*. Grundlehren der Mathematischen Wissenschaften [Fundamental Principles of Mathematical Sciences], Vol. 275. Springer-Verlag, Berlin, Germany. vii+352 pages. Fourier integral operators.

[20] IEEE 2004. IEEE TCSC Executive Committee. In *Proceedings of the IEEE International Conference on Web Services (ICWS '04)*. IEEE Computer Society, Washington, DC, USA, 21–22. doi:10.1109/ICWS.2004.64

[21] Markus Kirschmer and John Voight. 2010. Algorithmic Enumeration of Ideal Classes for Quaternion Orders. *SIAM J. Comput.* 39, 5 (Jan. 2010), 1714–1747. doi:10.1137/080734467

[22] Donald E. Knuth. 1997. *The Art of Computer Programming, Vol. 1: Fundamental Algorithms (3rd. ed.)*. Addison Wesley Longman Publishing Co., Inc.

[23] David Kosiur. 2001. *Understanding Policy-Based Networking* (2nd. ed.). Wiley, New York, NY.

[24] Leslie Lamport. 1986. *LaTeX: A Document Preparation System*. Addison-Wesley, Reading, MA.

[25] Newton Lee. 2005. Interview with Bill Kinder: January 13, 2005. Video. *Comput. Entertain.* 3, 1, Article 4 (Jan.-March 2005). doi:10.1145/1057270.1057278

[26] Dave Novak. 2003. Solder man. Video. In *ACM SIGGRAPH 2003 Video Review on Animation theater Program: Part I - Vol. 145 (July 27–27, 2003)*. ACM Press, New York, NY, 4. doi:99.9999/woot07-S422 http://video.google.com/videoplay?docid=6528042696351994555

[27] Barack Obama. 2008. A more perfect union. Video. Retrieved March 21, 2008 from http://video.google.com/videoplay?docid=6528042696351994555

[28] Poker-Edge.Com. 2006. Stats and Analysis. Retrieved June 7, 2006 from http://www.poker-edge.com/stats.php

[29] R Core Team. 2019. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. https://www.R-project.org/

[30] Bernard Rous. 2008. The Enabling of Digital Libraries. *Digital Libraries* 12, 3, Article 5 (July 2008). To appear.

[31] Mehdi Saeedi, Morteza Saheb Zamani, and Mehdi Sedighi. 2010. A library-based synthesis methodology for reversible logic. *Microelectron. J.* 41, 4 (April 2010), 185–194.

[32] Mehdi Saeedi, Morteza Saheb Zamani, Mehdi Sedighi, and Zahra Sasanian. 2010. Synthesis of Reversible Circuit Using Cycle-Based Approach. *J. Emerg. Technol. Comput. Syst.* 6, 4 (Dec. 2010).

[33] Joseph Scientist. 2009. The fountain of youth. Patent No. 12345, Filed July 1st., 2008, Issued Aug. 9th., 2009.

[34] Stan W. Smith. 2010. An experiment in bibliographic mark-up: Parsing metadata for XML export. In *Proceedings of the 3rd. annual workshop on Librarians and Computers (LAC '10, Vol. 3)*, Reginald N. Smythe and Alexander Noble (Eds.). Paparazzi Press, Milan Italy, 422–431. doi:99.9999/woot07-S422

[35] Asad Z. Spector. 1990. Achieving application requirements. In *Distributed Systems* (2nd. ed.), Sape Mullender (Ed.). ACM Press, New York, NY, 19–33. doi:10.1145/90417.90738

[36] Harry Thornburg. 2001. *Introduction to Bayesian Statistics*. Retrieved March 2, 2005 from http://ccrma.stanford.edu/~jos/bayes/bayes.html

[37] TUG 2017. *Institutional members of the TeX Users Group*. Retrieved May 27, 2017 from http://wwtug.org/instmem.html

[38] Boris Veytsman. 2017. *acmart—Class for typesetting publications of ACM*. Retrieved May 27, 2017 from http://www.ctan.org/pkg/acmart

A RESEARCH METHODS

A.1 Part One

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Morbi malesuada, quam in pulvinar varius, metus nunc fermentum urna, id sollicitudin purus odio sit amet enim. Aliquam ullamcorper eu ipsum vel mollis. Curabitur quis dictum nisl. Phasellus vel semper risus, et lacinia dolor. Integer ultricies commodo sem nec semper.

A.2 Part Two

Etiam commodo feugiat nisl pulvinar pellentesque. Etiam auctor sodales ligula, non varius nibh pulvinar semper. Suspendisse nec lectus non ipsum convallis congue hendrerit vitae sapien. Donec at laoreet eros. Vivamus non purus placerat, scelerisque diam eu, cursus ante. Etiam aliquam tortor auctor efficitur mattis.

B ONLINE RESOURCES

Nam id fermentum dui. Suspendisse sagittis tortor a nulla mollis, in pulvinar ex pretium. Sed interdum orci quis metus euismod, et sagittis enim maximus. Vestibulum gravida massa ut felis suscipit congue. Quisque mattis elit a risus ultrices commodo venenatis eget dui. Etiam sagittis eleifend elementum.

Nam interdum magna at lectus dignissim, ac dignissim lorem rhoncus. Maecenas eu arcu ac neque placerat aliquam. Nunc pulvinar massa et mattis lacinia.