Information Theoretical Optimal Use of RF Side Channels for Microsystem Characterization

Isaac Dekine\textsuperscript{a}, Mark Chilenski\textsuperscript{a}, Corey Cleveland\textsuperscript{a}, Piyush Kumar\textsuperscript{a}, Gil Raz\textsuperscript{a}, and Max Li\textsuperscript{a}

\textsuperscript{a}Systems & Technology Research, 600 W Cummings Park, Woburn, MA 01801

\textbf{ABSTRACT}

The use of involuntary analog side-channel emissions to remotely identify the internal state of digital platforms has recently emerged as a valuable tool in the arsenal of defensive measures against intrusion and malicious attacks, as well as hardware modifications. In particular RF emissions have been shown to be effective in this task. One of the key challenges is identifying and selecting useful features from the noisy signals which simultaneously enable the detection of the internal digital state reliably while minimizing the complexity of this operation. Our team has developed such sensors and we show the ability to optimally select features as well as optimally select bands of operation from which features can be drawn. Optimality here is in the sense of maximizing the mutual information between the features and the true state of the devices under test. In addition to being optimal in the sense of performance and low complexity for the real-time operation, the process of finding the optimal features is parsimonious and amenable to deployment in adaptive real-time sensors. In these proceedings we describe specific examples related to the detection of intended vs unintended programs on IoT devices and FPGAs as well as identification of other internal device settings. We show near-perfect identification of such internal states, achieved in real-time at distances of several feet in challenging environments.

\textbf{Keywords:} Cybersecurity, information theory, machine learning, vulnerability detection, cyber sensing, analog side-channel, RF measurements, internet of things

\section{1. INTRODUCTION}

Embedded computing platforms have become ubiquitous and an integral part of many industries and applications. These range from Internet of Things (IoT) devices, through industrial control systems, to FPGA subsystems to name a few. The reliance on such platforms, in sensitive industries such as defense, civilian infrastructure, health, automotive, and telecommunications, makes the ability to trust them imperative. The globalized and distributed nature of electronic device design, manufacturing, and integration gives rise to serious concerns over supply-chain integrity. Therefore, a thorough identification and characterization of the internal state of such devices is crucial. These include identification of device identity, internal hardware and software settings and modifications, and connection to peripherals for devices that have been fielded as well as in the supply chain without requiring disruption to ongoing operations.

Multiple methods exist for authentication and characterization of systems which have internally enough memory and computational resources, such as wireless networking systems and client desktop computers. These typically utilize methods in the upper Open Systems Interconnect (OSI) model layers and/or employ host-based intrusion detection systems (IDS), often in a sandboxed environment. However, these methods are not readily applicable to embedded devices which have no network connection and/or do not have sufficient computational resources. In this paper, we outline a principled approach to characterize the internal state of embedded devices across an air-gap using side-channel information in the form of their involuntary radio-frequency (RF) emissions during operation. Because this approach receives and processes the RF emissions using a completely separate system, it is not constrained by the physical and cyber limitations of the device under test (DUT).

RF attributes have been previously used for device identification, authentication, and operational status assessment.\textsuperscript{1–3} However, they have limitations in general. Many such studies require expensive and elaborate
sensor platforms and/or require the RF probes to be very close to the DUT, which may not be feasible in many situations. In addition, most of these studies do not leverage the information available in RF signals in an optimal manner for an in-depth characterization of the internal state of the embedded device. One key limitation of existing methods is the complexity of selecting a computationally tractable subset of RF signals which has information about the internal DUT state. In contrast, the method we describe provides a principled information-theoretic optimal approach to analyzing and exploiting available information from RF channels for a given system setup - minimizing complexity while retaining maximal performance. For other recent work in this direction, see References 4–6.

The rest of this paper is organized as follows. In the rest of the Introduction, we describe the system setup (Section 1.1) and the motivation for using information theoretic methods in this context (Section 1.2). Sections 2.1 and 2.2 explain the mathematical concepts behind mutual information optimization and how this is used for feature selection and operational band selection. In Section 3 we describe measured experiments that demonstrate the value of these results. Finally, in Section 4 we discuss and summarize these results and outline possible next steps.

1.1 The LASSES System Setup

Figure 1: LASSES System Block Diagram: The LASSES system consists of two components: i) an offline training mode in which a set of different DUT configurations are selected and measured, and for which features are selected and models are trained, and ii) a fielded operation mode, in which the optimally informative features and models are used in real-time to identify the internal state of the DUT. The focus of this paper is the “Signal Isolation, Mutual Information Calculation, & Feature Analysis” block in the offline component.

Our system, known as “Leveraging Anomalous System Signals to Enable Security” (LASSES), is designed to obtain information about the internal states of electronic DUT’s by measuring involuntary analog side channel signals from them.

Figure 1 provides a high-level description of our system. The system comprises two main components. The first is an offline training component that captures repeatable, high-fidelity measurements of a device’s side channel emissions. It performs signal processing and analysis to extract relevant features about the internal state of the DUT. We note that while we focus here on electromagnetic side channels, our approach is independent of the analog modality and hence can be extended to include acoustic, thermal, IR, or power usage signals as well as intended emissions if available. The focus of this paper is the “Signal Isolation, Mutual Information Calculation, & Feature Analysis” block in Figure 1, which consists of processing the measured data and applying parsimonious but effective methods to extract the most useful and discriminative information about the internal state of the device. Optimality of features here is in the sense of their value for the overall mission of “Identification of Processor State / Mode”.

2
The insights obtained from the offline training component are used to inform the second component, which works in a real-time fielded environment. This component is designed to receive analog emissions and perform adaptive signal processing and feature selection, followed by state estimation and tracking using models developed with the offline component. This allows the system to identify the internal state of the DUT as requested by the user for a given set of measurement parameters.

1.2 Information Theoretic Methods for Device Characterization Using Involuntary RF Emissions

Involuntary emissions from embedded devices come in various forms, most of which are weakly radiated into the far field. Among other things, the weak radiative quality is generally a consequence of the physical size and shape of the computing devices, and is not ideal for electromagnetic radiation at most frequencies. Furthermore, the designers of these devices are also constrained by the electromagnetic compatibility (EMC) requirements of the countries in which they are sold, leading hardware designers to ensure that RF emissions are below allowable limits. There are many well-known and commonly employed methods for designing devices to minimize these emissions,\(^7\) which makes the task of sensing device emissions in the far field non-trivial. In addition, external interference signals, either from unintentional emissions of other devices or from intentional emissions such as communications, can be stronger than the signals radiating from a device of interest. Finally, an important issue to take into account is that many of the signals that contain little or no information are often significantly stronger than those that are informative.

All of these factors introduce a need to be able to rapidly separate out the received signals at a particular time and band of the RF spectrum, and determine which portions of this signal are informative relative to the internal state of the device that we are interested in characterizing. It should be clear that simply looking at the strongest or the most prominent signals will yield subpar results, as the strongest signals may not be the most informative. Indeed, they may be from some device other than the one we are interested in. This leads to the challenge of finding informative signals that may be significantly weaker than the uninformative signals. Rather than relying on heuristic feature selection methods such as signal magnitude or variance, we solve this issue by carrying out a systematic information theoretic analysis which extracts features with maximal information about a given internal device state. In particular, we compute the mutual information between various signal features and device states and optimally select the portions of the RF spectrum which maximize this mutual information.

2. BACKGROUND AND TECHNICAL DETAILS

2.1 Overview of Mutual Information for Feature Selection

We quantify the information content of a given feature or band in terms of the mutual information \(I(X;Y)\) between the measured signal \(X\) and the program identity (or property of interest) \(Y\). In machine learning parlance, \(X\) corresponds to the features and \(Y\) corresponds to the class labels. Mutual information is an information-theoretic quantity representing how much information random variable \(X\) conveys about random variable \(Y\) (and vice versa, as \(I(Y;X) = I(X;Y)\)).\(^8\) The mutual information of two discrete random variables is defined as

\[
I(X;Y) = \sum_{y \in \mathcal{Y}} \sum_{x \in \mathcal{X}} p(x, y) \log_2 \left( \frac{p(x, y)}{p(x)p(y)} \right),
\]

(1)

where \(\mathcal{X}\) and \(\mathcal{Y}\) are the sample spaces of \(X\) and \(Y\), respectively, and the base of the logarithm determines the units (bits in this case). Note that this makes no assumption about the dimensionality of \(X\) and \(Y\). For instance, \(X\) could be a tuple containing the power at each frequency bin in a given RF band and \(Y\) could be a tuple of the currently executing program and the identity of the device executing it. The advantage of mutual information over simpler quantities like covariance is that it completely captures the effects of multiple variables, even if they interact in a nonlinear way. In particular,

\[
I(\{X_1, X_2\};Y) \leq I(X_1;Y) + I(X_2;Y),
\]

(2)
because $X_1$ and $X_2$ may have redundant information about $Y$. Mutual information is bounded by $0 \leq I(X; Y) \leq H(Y)$ (where $H(Y) = -\sum_{y \in Y} p(y) \log_2 p(y)$ is the entropy in $Y$; this represents the uncertainty in $Y$ absent any measurements), where the lower bound corresponds to the case where $X$ and $Y$ are independent, and the upper bound corresponds to the case where $Y$ is completely determined by $X$.

The (weak) Fano inequality sets a lower bound on the probability that the program identity $Y$ is misclassified:

$$P_e \geq \frac{H(Y) - I(X; Y) - 1}{\log_2 |Y|},$$

where $P_e$ is the probability of mis-classification and $|Y|$ is the number of classes. Therefore, to minimize the probability of errors, it is necessary to find the $X$ which maximizes the mutual information.

Specifically, given the set of possible features $\Omega$, we would like to pick the $k$ features $X \subset \Omega$ which maximize $I(X; Y)$:

$$X_{opt} = \arg \max_{|X| \leq k} I(X; Y).$$

In general, this is a combinatorial search problem, requiring exhaustive enumeration of all $O(2^{|\Omega|})$ subsets of features with $|X| \leq k$. Mutual information, however, has the property that it is submodular:

$$I(X_1; Y) + I(X_2; Y) \geq I(X_1 \cup X_2; Y) + I(X_1 \cap X_2; Y), \quad \text{for } X_1, X_2 \subseteq \Omega.$$

A key implication of this property is that an $O(k|\Omega|)$ greedy algorithm which adds the most informative feature at each step and never removes features will provide a near-optimal solution, within a factor of $1 - (k-1)/k \sim (e-1)/e$ of the true maximum. For submodular set functions, the intractable combinatorial search can be efficiently approximated by a tractable polynomial time greedy search. Henceforth, the word “optimal” will be taken to refer to the near-optimal subset obtained in this manner.

In practice, $I(X; Y)$ is estimated using the Kozachenko-Leonenko estimator applied to measured, labeled data. Because the accuracy of this estimator breaks down in a complicated way as more features are added (given a fixed number of measurements), it is often impractical to compute the mutual information for more than a few dozen features. Instead, to estimate the information content of a given band (which may contain hundreds of frequency bins, many of which may be redundant and/or uninformative), we optimally add features from the band until the estimated mutual information starts to decrease even for the “most informative” feature to add. Because $I(\{X_1, X_2\}; Y) \geq I(X_1; Y)$, a decrease in $I(X; Y)$ when the “best” feature is added to $X$ is a clear sign that the remaining features are redundant and we can use the lower-dimensional estimate as an approximation to the full value. In the experiments presented here, this rollover tends to occur after around 10 features are added, so the experiments in Section 3 use the information of the top 10 features to quantify the information in a given band.

### 2.2 Using Mutual Information to Determine Optimal Frequency Bands and Other Parameters

As described in Section 2.1, we can estimate the mutual information content contained by a near-optimal subset of features $X_{opt}$. This estimate is intrinsically useful, as it allows us to understand the limit on our ability to classify similar data. However, in a real settings other limiting factors may need to be accounted for. Examples include the ability to limit sensor size and performance by selecting a relatively small band of operation rather than using optimal features from across the spectrum. Such sub-selection is preferably quantified relative to the overall available information-theoretic limits. If multiple narrow bands are equally informative then these practical and quantifiable parameter selection steps also allow for online adaptation to changing signal environment.

Note that in practice, mutual information estimate of features applies to the parameters of the data quantified; for instance, measurements taken using a particular log periodic antenna from a distance of 2 feet away, and captured using an RF receiver with 10 MHz of instantaneous bandwidth centered at 530 MHz. This mutual information estimate is particular to the specifics of the measurement and any processing that yielded data $X$,
as well as the class labels $Y$. However, extending this quantification can also inform us regarding the information content relative to these specific parameters. First, we perform multiple measurements, for instance, at different distances between the sensor and the DUT, or at multiple center frequencies. We then quantify the mutual information content of the optimal features for each of the measurements independently.

The results of this framework provide us with two major advantages. First, it allows us to examine and understand the relationship that the parameters have on the observable mutual information between the class labels, such as the impact of distance between the DUT and our system. Second, it allows us to determine the bounds on system design, relative to the parameters that we can design for. For example, knowing that we can maximize the mutual information using 1 MHz of receiver bandwidth instead of 100 MHz allows for real-time system designs that utilize lower bandwidth (and consequently less expensive) receivers and reduced data processing components, which ultimately allow for smaller, cheaper, and more efficient system designs without sacrificing performance.

In this paper, we focus on mutual information scanning to solve a common problem: determining the particular portions of the spectrum (the RF band) that contain the most mutual information relative to the class labels. In addition, we will be interested in quantifying how much of that band is necessary for attaining this optimal information content. In this manner, the heretofore cumbersome process of manually attempting to identify informative portions of spectrum via imperfect metrics such as signal-to-noise ratio (SNR) can be automated and improved in a reliable manner.

3. EXPERIMENTS AND RESULTS

![Experimental setup for measuring device emissions using the LASSES system](image)

Figure 2: Experimental setup for measuring device emissions using the LASSES system: The LASSES system, composed of a log-periodic antenna connected to a software-defined radio for data collection is shown on the right of the image. The device under test, an RT5350 in this photo, is shown to the left. Both the LASSES system and device under test are placed inside of a shielded, electromagnetically anechoic chamber to reduce the impact of outside RF emitters. All measurements described were conducted using this experiment setup.

In this section, we describe the setup used for measured experiments, followed by a discussion of four experiments that show the value of the mathematical framework described above.

3.1 Experimental Setup

The LASSES sensor has three main components: a log periodic antenna, a software-defined radio, and a control laptop. The system can be seen in Figure 2, not pictured is the control laptop which is sitting outside of the anechoic chamber. The log periodic antenna and software-defined radio comprise the measurement suite of the LASSES system as detailed in Figure 1. After being collected by the software-defined, radio data is transmitted to the control laptop. The laptop handles all feature generation and mutual-information calculations, including our mutual information scanning algorithm, and is the last step in the offline training.
Figure 3: Mutual information relative to program labels for the Arduino Uno device. The mutual information relative to three program labels is shown as a function of the sub-band start frequency for two bandwidths, 10 MHz (blue) and 1 MHz (red). The three programs studied are: \{Bubble sort, LCD blink controller, FFTW\}, described in Section 3.2. The mutual information is computed relative to program labels for the top ten discriminative frequency bins based on the amplitudes in each bin using the methods described in Section 2.1. The harmonics of the Arduino clock are indicated by the black vertical lines at the top of the figure. Both sub-bands with widths 10 MHz and 1 MHz provide maximal mutual information close to the program label entropy (shown as a dashed black horizontal line), as long as they contain a clock harmonic frequency. This is because the clock harmonics provide full information about the running program (also see Figure 4).

For each experiment described below, the DUT is set up approximately two feet away from the LASSES system. The DUT runs a program repeatedly in a loop, while the LASSES system scans across 400-1000 MHz. This scanning is performed using 100 MHz of instantaneous bandwidth, thus the center frequencies actually measured are 450, 550, 650, 750, 850 and 950 MHz. For each band, the system records one second of emissions and decomposes them into raw data features for that program and frequency sub-band. After the completion of RF-band scanning for one program, the same process is repeated for all other DUT programs. All measurements were conducted inside a shielded anechoic chamber. Similar data was collected in unshielded settings and performance is similar, however interference artifacts are introduced. The data collected for a set of programs is used to compute Fourier domain features with an analysis window of 100,000 complex samples, resulting in a bandwidth resolution of 1 KHz. The features are then separated into two different sub-bands: one of width 1 MHz with no overlap, and another with width 10 MHz with 90 percent overlap. The motivation for using two sub-bands with different widths is to determine if sufficient discriminative information is present in them. We will see later that the amount of useful information will depend on the nature of the DUT and the internal state of the devices for which knowledge is desired.

Finally, using the methods described in Section 2.1, features which maximize mutual-information are selected in each band. This is done for both bandwidths, 1 MHz and 10 MHz, around each center frequency. Since the mutual information of the top ten features relative to the program labels provides a good approximation to the total maximal mutual information in a band, we use this quantity to present our results for each experiment.

3.2 Experiment 1: Distinguishing Between Different Programs on Arduino Uno

In this experiment, three programs are run on the Arduino Uno Microcontroller device: \{bubble sort, LCD screen blink controller, & FFTW\}. The bubble sort program sorts a fixed length array of random integers generated using the device’s random number generator. The LCD screen blink controller blinks a cursor on a screen
Figure 4: Averaged emission spectra for three programs for the Arduino Uno device near 528 MHz. The 528 MHz harmonic of the Arduino clock is displayed in the average emission spectra for each of the three programs: (i) Bubble sort (blue), (ii) FFTW (red), (iii) LCD Blink controller (yellow). The frequency of the clock clearly varies in a manner that is dependent on the program running on the device, indicating that full information about the program can be obtained from the clock signal itself. This provides a confirmation of the results in Figure 3.

Connected via the on-board general purpose I/O (gpio) pins, however no screen is connected over gpio during the tests. The FFTW algorithm takes in an array of the same fixed length of random integers as the bubble sort program and computes its fast Fourier transform (FFT) coefficients. For all of these tests, the Arduino device is clocked at 16 MHz using its onboard ceramic resonator.

The computed mutual information for the features with respect to the programs discussed above is shown in Figure 3 for both the 1 and 10 MHz sub-bands. The figure clearly shows that maximal mutual information, nearly reaching the program label entropy $H(Y)$ (see Section 2.1), is obtained in any band that contains one of the 16 MHz clock harmonics. These clock harmonics are indicated by black vertical lines at the top of the plot. In this case, the bandwidth of each sub-band has little impact on the information content of the band. The important requirement is whether or not the band (with width 1 MHz or 10 MHz) contains a clock harmonic. The above indicates that the clock signal itself is the strongest source of information in the emission spectra. This hypothesis is confirmed by the plot in Figure 4, which shows the clock signal in the averaged emission spectra of data collected for each of the three programs for the Arduino Uno. The figure clearly shows that the frequency of the clock signal varies with the program on the device, thus enabling the clock signal itself as an informative feature.

### 3.3 Experiment 2: Distinguishing Between Different Programs on RT5350

In this experiment, four programs are measured on an RT5350F-OLimuXino development board (RT5350): \{FFT, prime number calculation, gzip, & hashing (md5sum)\}, as well as device idle (no programs running). The RT5350 is built around a MIPS24KEc processor and runs the OpenWrt Linux operating system. The FFT program calculates the Fourier coefficients of a fixed length array of random integers. The prime number calculation program iterates a fixed number of times, and on each iteration checks if the iteration counter is a prime number. The gzip program compresses a file of randomly generated data from the pseudorandom number generator included in OpenWrt. Finally, the md5sum program computes the hash of a file generated from the pseudorandom generator included in OpenWrt. The clock speed of the processor is 360 MHz, which is derived from a 20 MHz crystal oscillator located on the board. To execute each program, a host computer is connected.
Figure 5: Mutual information relative to program labels for the RT5350 device. The mutual information relative to five program labels (one of them being the idle state) is shown as a function of the sub-band start frequency for two bandwidths, 10 MHz (blue) and 1 MHz (red). The programs studied are: \( FFT, \) prime number calculation, \( gzip, \) hash sum (md5sum), and idle (no programs running), described in Section 3.3. The mutual information is computed relative to program labels for the top ten discriminative frequency bins based on the amplitudes in each bin using the methods described in Section 2.1. The harmonics of 120 MHz that are used to generate the 360 MHz clock are indicated by the black vertical lines at the top of the figure. Both sub-bands with widths 10 MHz and 1 MHz provide maximal mutual information close to the program label entropy \( H(Y) \) (shown as a dashed black horizontal line), as long as they contain a 120 MHz harmonic. Also see Figure 6 which shows that mutual information relative to program labels is contained in the inter-modulations caused by the programs around the 120 MHz harmonics.

The computed mutual information for the features with respect to the programs discussed above is shown in Figure 5 for both the 1 and 10 MHz sub-bands. Similar to that for the Arduino Uno, the mutual information in both of these sub-bands reaches the maximal value (equal to the program label entropy \( H(Y) \)) at the 720 MHz clock harmonic. However, maximal mutual information can also be obtained in both sub-bands that contain harmonics of 120 MHz. These harmonics are denoted by black vertical lines in Figure 5. We think that this is due to the fact that the underlying crystal oscillator at 20 MHz that sources the clock also leaks information at other frequencies that are multiples of 20 MHz. The high mutual information values found at other harmonics of 20 MHz support this hypothesis. For reasons that we have not yet explored, not all 20 MHz harmonics provide information, however the 120 MHz frequency and its harmonics are particularly informative. Figure 6 shows the 720 MHz harmonic of the device clock in the averaged emission spectra for each of the four programs and the idle state. In contrast to the Arduino Uno, the RT5350 device has a much more stable clock that does not deviate with the programs. However, as can be confirmed from the figure, different programs cause different inter-modulations around the 120 MHz harmonics (including the clock harmonics) which provide discriminative information regarding program execution.

3.4 Experiment 3: Distinguishing Between Different Programs on FPGA Soft Processor

For this experiment, the DUT is the Arty development board, which contains a Xilinx Artix 7 FPGA. The FPGA is configured to run as a Microblaze soft core processor with four different programs: \{merge sort, insert sort,
Figure 6: Averaged emission spectra for five programs for the RT5350 device near 720 MHz: The 720 MHz harmonic of the device clock (and of 120 MHz) is displayed in the average emission spectra for each of the five programs: (i) idle (blue), (ii) FFT (red), (iii) prime number calculation (yellow), (iv) gzip (purple) and (v) hash sum (green). The figure clearly shows that the differences in the inter-modulations around the 120 MHz harmonics (the 720 MHz shown here) caused by different programs contain most of the mutual information relative to program labels.

Figure 7: Mutual information relative to program labels for the Arty FPGA device: The mutual information relative to four program labels is shown as a function of the sub-band start frequency for two bandwidths, 10 MHz (blue) and 1 MHz (red). The programs studied are: {quick sort, merge sort, insert sort, and bubble sort}, described in Section 3.4. The mutual information is computed relative to program labels for the top ten discriminative frequency bins based on the amplitudes in each bin using the methods described in Section 2.1. The mutual information in this case is not concentrated around individual harmonics. Rather, there exists two wide regions with high mutual information: 400-550 MHz and 700-850 MHz. Also, the sub-band with width 10 MHz provides more information relative to that with width 1 MHz.
Figure 8: **Averaged emission spectra for four programs for the Arty FPGA device in the range 400-1000 MHz**

The average emission spectra for each of the five programs, (i) Quick sort (blue), (ii) Merge sort (red), (iii) Insert sort (yellow), (iv) Bubble sort (purple), clearly shows the wide band nature of the emissions from the Arty FPGA board with an drop in power in the 600-700 MHz range correlating with the drop of mutual information in that range, see Figure 7.

Figure 9: **Mutual information relative to program labels for the Arty FPGA device**

The mutual information relative to four compiler optimization flags for a given program, Merge sort, is shown as a function of the sub-band start frequency for two bandwidths, 10 MHz (blue) and 1 MHz (red). The compiler options flag used are: {O0, O1, O2, O3}. The mutual information is computed relative to program labels for the top ten discriminative frequency bins based on the amplitudes in each bin using the methods described in Section 2.1. The mutual information in this case is not concentrated around individual harmonics, similar to that in Experiment 3. The mutual information is distributed over a wide region with a high-information region at 400-600 MHz. Also, the sub-band with width 10 MHz provides more information relative to that with width 1 MHz.

*quick sort, & bubble sort*. Each sorting algorithm is run in a continuous loop where it is provided a fixed length
array of random integers. While the clock speed of the Arty device is provided by a 100 MHz crystal oscillator, the FPGA uses this clock to create a multitude of other clocks that are in use on the board for both its soft core processor as well as data transmission. Due to this, the device does not have a single identifiable clock speed like the RT5350 and Arduino Uno.

The lack of a single clock frequency gives rise to two main differences in the pattern of mutual information relative to program labels for the Arty FPGA when compared to the Arduino Uno and RT5350 devices. The first is that the bulk of the mutual information relative to program labels is not solely based on the harmonics of a particular frequency, as can be seen in Figure 7. Instead, there are two regions with high mutual information content: 400-550 MHz and 700-850 MHz. The second difference is that now the sub-band with bandwidth of 10 MHz provides more information compared to the sub-band with width 1 MHz. These results are supported by the plot in Figure 8, which clearly shows that the emission spectra for programs from the Arty FPGA has a much wider band in contrast to those for the Arduino Uno and the RT5350 devices.

3.5 Experiment 4: Distinguishing Between Same Program Compiled Differently on FPGA Soft Processor

For this experiment, the Arty FPGA is configured as a Microblaze soft core processor running the Merge sort program. The program is compiled with four different optimizations: \{O0, O1, O2, O3\}. As can be seen from Figure 9, the mutual information relative to compiler optimization flags is distributed over a wide range of frequencies similar to that in the previous experiment. However, now the region with lower frequencies, 400-600 MHz, contains a large fraction of the mutual information relative to optimization flags. Again, the averaged emission spectra for the different compiler optimization flags shown in Figure 10 confirms this result.

4. DISCUSSION, SUMMARY, AND NEXT STEPS

In this paper, we described a systematic and robust framework for measuring involuntary analog side-channel emissions from a variety of electronic devices and extracting useful information from these signals to detect the internal states of these devices. We presented a technique for using information theory to systematically and efficiently characterize an electronic device’s signals and discover portions of those signals that are near-optimal in detecting the internal states of the device.
We presented multiple results employing these techniques on real measured data on three classes of devices: Arduino Uno, RT5350 and Arty FPGA, and demonstrated their efficacy in identifying discriminative features from emissions that yield near-perfect identification of their internal states. Internal states that we studied in this paper include the particular programs running on the devices (Arduino Uno, RT5350 and Arty FPGA), and the compiler optimization flags for a given program running on the Arty FPGA device.

These results have important implications. First, they show that security monitoring of fielded systems via their involuntary side-channel emissions is possible by parsimonious data processing with only a few features, without sacrificing performance. This allows for decreased system requirements and greater scalability. Second, since our methods quantify the mutual information relative to internal device state for different bands, it allows the system to quickly identify the next most informative band if a given band becomes unavailable or unusable due to some reason, such as noise or interference. Finally, they provide a better understanding of the kind of features that provide the most information relative to an internal state of a given DUT. For instance, we saw that the movement of the clock frequency provided the most information about programs for the Arduino Uno, while inter-modulations around the clock harmonics were the most informative features for the RT5350 device. By contrast, the Arty FPGA had multiple relevant clocks and consequently the most informative features are not related to any one clock frequency (or its harmonics).

Significant future work related to these methods is being considered by our team, including:

- Using mutual information scanning methods to further explore relationships between all LASSES system parameters and performance.
- Exploring analog modalities other than EM propagation and quantifying their relative information content.
- Understanding the robustness and limitations of mutual information estimates in the presence of adverse factors such as non-stationary noise and interference, and creating a framework to model and automate the process of avoiding such concerns.

Acknowledgments

The authors wish to thank the Air Force Research Laboratory (AFRL) and the Defense Advanced Research Projects Agency (DARPA) for funding this work as part of the LADS program. In particular we thank Mr. Ian Crone and Dr. Angelos Keromytis, the present and former DARPA program managers of LADS, for their encouragement and support throughout the program.

This research was developed with funding from the Air Force Research Laboratory (AFRL) and the Defense Advanced Research Projects Agency (DARPA). The views, opinions and/or findings expressed are those of the author and should not be interpreted as representing the official views or policies of the Department of Defense or the U.S. Government.

REFERENCES


