ECUPrint - Physical Fingerprinting Electronic Control Units on CAN Buses inside Cars and SAE J1939 Compliant Vehicles

Lucian Popa, Bogdan Groza, Camil Jichici and Pal-Stefan Murvay

***Abstract*—We fingerprint 54 ECUs from 10 cars, one of them being a heavy-duty vehicle that is compliant to the SAE J1939 standard. These later specifications implemented in commercial vehicles offer concrete sender addresses in every CAN frame, making physical characteristics easier to link to specific ECUs. This is not the case for traffic collected inside passenger cars where the allocation of CAN bus identifiers is non-uniform, without explicit sender and receiver addresses, making ECU identification more challenging. While previous research has shown good separation between ECUs even when single features are used, e.g., skews or maximum voltage level, prior results are based on a small number of cars, while our larger experimental basis proves that single features are likely insufficient to separate between a large number of ECUs. Concretely, for a crisp separation, at least four features seem to be needed, i.e., mean voltage, max voltage, bit time and plateau time, while clock skews or any single voltage feature lead to overlaps. We provide clear experimental bounds on the intra and inter-distances regarding skews and voltage features, not neglecting environmental variations which may occur when the car is running.**

***Keywords*-CAN bus, clock skews, ECU fingerprinting, J1939**

I. Introduction and motivation

The CAN bus has more than a decade of reported security vulnerabilities [1], [2], [3]. This doesn't come as a surprise for a bus designed in the 80s. In fact, its survival for so many decades is a strong argument to support its future use, despite some obvious security shortcomings. Moreover, the recent introduction of the higher speed CAN-FD makes it clear that CAN will be present inside cars for the decades that follow. The larger payload of CAN-FD will make it much easier to accommodate cryptographic material inside frames responding to modern security needs. Physical fingerprinting techniques have a complementary role, e.g., for authentication or forensics, which is not going to be ruled out by cryptographic security. There are many incidents which prove this is so. For example, by corrupting an existing ECU (Electronic Control Unit), cryptographic keys can be extracted, as demonstrated in a recent attack by which diagnostic security keys were extracted from a real-world vehicle [4]. Other research works have shown that it is possible to compromise the central information unit even from remote [5] or have attacked ECUs via over-the-air update protocols [6]. Cryptographic keys may be also extracted by various side-channel attacks, these attacks are commonly reported in embedded systems [7] and have been also demonstrated on in-vehicle controllers [8], [9]. Testing vulnerabilities against such attacks is now a required practice in the automotive industry [10]. Consequently, physical fingerprinting technologies are here to stay as an additional layer of protection besides cryptographic security.

Potential use cases. Our work goes mostly in the same vein as the work from [11] which attempts to map the existing ECUs inside the vehicles based on physical characteristics. However, the authors in [11] rely exclusively on clock characteristics which prove to be unsatisfactory in our analysis over a larger experimental basis due to obvious overlaps between ECUs in distinct cars. We use clock skews and, more importantly, several voltage features to test our methodology on a large experimental basis: 10 cars containing 54 ECUs. One of the most common use cases for physical fingerprinting is the development of intrusion detection systems, we enumerate several works in this direction in the related work section. However, in this work we are mostly interested in the extraction of such fingerprints to uniquely identify the ECUs inside a vehicle for forensics purposes. It is worrisome that, according to recent data from the 2020 FBI crime statistics report[1], car theft has increased in recent years. Moreover, a new concern has emerged in the context of vehicle identity theft: *VIN cloning*. By this attack, the vehicle identification number (VIN) of an existing car is cloned. A report from the National Crime Prevention Council[2] shows that many stolen vehicles also rely on cloned VINs. Physical fingerprints may help to alleviate such problems as they can be stored in authorized databases and inspected/updated as the vehicle goes to standard procedures such as the annual (or biennial) technical safety inspection which is mandatory all over the world. No less, such fingerprints may be checked during regular traffic inspections, which are even more common for heavy duty vehicles (SAE J1939 compliant). While it is true that these fingerprints may vary over time, it is expected that the periodic collection of such fingerprints will offer better clues on how the fingerprints will vary, giving precious hints for forensics and authentication purposes. Ultimately, our work is concerned in showing the intra and inter-distances between such fingerprints over a large number of ECUs, i.e., 54 from

Lucian Popa, Bogdan Groza, Camil Jichici and Pal-Stefan Murvay are with the Faculty of Automatics and Computers, Politehnica University of Timisoara, Romania, Email: {lucian.popa, bogdan.groza, camil.jichici, pal-stefan.murvay}@aut.upt.ro

[1]https://www.fbi.gov/news/pressrel/press-releases/fbi-releases-2020-crime-statistics

[2]http://archive.ncpc.org/resources/files/pdf/celebrate-safe-communities/NCPC-autotheft-101.pdf

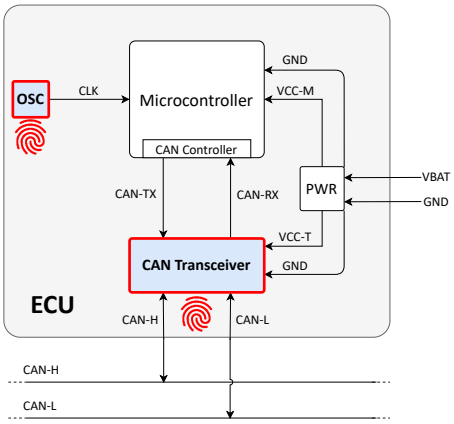


Fig. 1. Internal block diagram of an automotive ECU with architectural components required for CAN communication

10 cars, which was not considered by any of the previous works (which are generally experimenting with 2 or 3 cars at most).

Sources for fingerprinting. Figure 1 depicts the physical components that exist inside an ECU which are needed to support its functions, including CAN communication. Automotive ECUs are supplied from the battery voltage VBAT and connected to the vehicle ground GND. Internal supplies for the microcontroller and CAN transceiver denoted as VCC-M and VCC-T are provided from the ECU power management module denoted in the figure as PWR. CAN frames are transmitted and received using the CAN-TX and CAN-RX serial lines by the CAN Controller usually embedded inside the Microcontroller. These frames are converted to and from differential voltages on the CAN-H (CAN High) and CAN-L (CAN Low) lines by a CAN Transceiver. An external oscillator OSC provides the clock signal CLK required by the Microcontroller to manage the internal operations allowing it to perform specific timing related actions, e.g., adjust CAN bit time or transmit periodic CAN frames, etc.

In the light of the above, there are two main physical characteristics that are exposed by ECUs on the CAN bus: i) clock skews which can be extracted from the timing of CAN frames and ii) voltages which can be extracted from the physical signal on the bus. Both these characteristics have been used by previous works to identify ECUs, although to the best of our knowledge no previous work has addressed them both in a single paper for the same or several vehicles. Table I provides a summary of the vehicles used in our experiments and the amount of data that has been collected from each of them. Our study focuses on 10 vehicles from which we sample more than two hundred thousand bits for voltage fingerprinting and collect more than eight million frames for skew computations that lead to the identification of 54 ECUs. Therefore, one of the first contributions that we exhibit is to provide comprehensive data on these two types of fingerprints for ECUs inside vehicles, ranging from personal cars up to heavy-duty vehicles that comply with the SAE J1939 standard. The advantage offered by heavy-duty vehicles is that the SAE J1939 standard provides clear information on the source of the messages and fingerprints can be directly linked to a specific sender. In case of passenger cars, such information is largely missing but we show that the CAN frames nicely cluster around specific ECUs and thus the fingerprint of each sender is easy to recognize.

A few words on the relevance of using both timing and voltage characteristics may be in order. Table II presents a brief summary on the weak and strong points for skews and voltages. Skews are easier to collect and they may be preserved when frames are retransmitted by gateways. This depends on the specific implementation, i.e., it will work in case when the retransmission is on-event, similar to the case of computers connected via gateways over large networks as previous works have shown that computers can be fingerprinted from remote [12] despite the fact that packets are running over multi hops in the network. If the gateway buffers the frames and transmits them using interrupts generated by the local timers, then indeed, the skew will not be preserved. Also, the frame arrival time allows the implementation of covert channels based on frame inter-arrival timings [13], [14]. The downside of skews is that they are easier to forge [15]. Moreover, they will not work for on-event transmissions and are affected by processing and arbitration delays. Extracting skews also requires large number of frames for correct estimation. On the other hand, voltage levels are harder to forge, they are feature rich and a single bit may be sufficient to recognize the sender. On the downside, they are harder to collect as they require high sampling rate ADCs and physical access to the bus. Because of this mixed image with pros and cons for both, it seems that using both timing and voltage fingerprints may help and we try to present them in a comparative manner in our work.

Summary of contributions. Briefly, the contributions of our work can be summarized as follows:

1) We collect comprehensive experimental data on skews and voltages from 10 vehicles, ranging from small cars to SUVs and a heavy-duty vehicle, totaling 54 ECUs. Our datasets will be made public and we hope that they can serve for future research works concerned with designing fingerprinting methodologies or intrusion detection systems based on physical characteristics.
2) From the collected data, we extract and analyze five features, partly used in previous research works, which can help to link each frame with a physical ECU and serve as a compact fingerprint for the ECU: clock skews, mean voltage, maximum voltage, bit and plateau time.
3) We present results on skews and the extracted voltage features in a comparative manner, showing concrete data on the intra and inter-distances, which are generally neglected in previous works, with respect to the aforementioned physical features for 54 ECUs.
4) We give a clear quantitative depiction on how these characteristics vary while two of the vehicles are operating for 1 hour.

The rest of the paper is organized as follows. Section II discusses the background on CAN, J1939 and related works while in Section III we show how data collection was performed and present the vehicles from our experiments. In Section IV

TABLE I
SUMMARY OF VEHICLES AND COLLECTED DATA*

Vehicle	Model year	No. ECUs**	No. IDs	Busload	Temperature	Battery voltage	Collected frames (skew)	Collected bits (voltage)
Honda Civic	2012-2017	6	43	31%	9 °C	12.8 V	1,039,512	40,073
Opel Corsa	2006-2014	4	29	23%	8 °C	13.9 V	442,992	9,187
Hyundai i20	2014-2020	7	40	35%	12 °C	14.2 V	616,296	17,767
John Deere Tract.	2010-2018	3	33	19%	4 °C	14.2 V	154,779	4,021
Dacia Duster	2010-2017	3	12	14%	10 °C	14.4 V	247,154	9,086
Dacia Logan	2012-2019	6	46	14%	10 °C	12.6 V	629,662	31,579
Hyundai ix35	2009-2015	6	26	45%	9 °C	13.5 V	847,161	23,104
Ford Fiesta	2017-2020	6	46	51%	5-7 °C	14.9 V	2,243,359	43,861
Ford Kuga	2013-2019	9	70	65%	9 °C	13.7 V	1,233,545	28,024
Ford Ecosport	2018-2021	4	87	43%	9 °C	15.0 V	759,421	22,808
Total		**54**	**432**				**8,213,881**	**229,510**

* the dataset is publicly released to serve for future investigations and can be retrieved from the ECUPrint project on GitHub and the authors institution server
** based on the fingerprinting methodology in this work

TABLE II
COMPARISON BETWEEN SKEW AND VOLTAGE FINGERPRINTING

	Advantages	Disadvantages
Skews	✓easy to collect ✓may be preserved through gateways (possible to retrieve from distinct buses)	✗easier to forge ✗do not work for on-event frames ✗affected by arbitration and processing delay ✗require many frames for estimation
Voltage	✓harder to forge ✓single bit/frame is sufficient ✓feature rich fingerprint	✗harder to collect, may require high sampling rate ADCs ✗require physical access to the same bus

we set up the theoretical framework for our analysis. Section V presents the results over the collected experimental data. Finally, in Section VI, we state the conclusions of our work.

II. BACKGROUND AND RELATED WORKS

In this section we provide some brief background on CAN and the J1939 upper layer implementation that is present in one of our vehicles, then we account the related works.

A. Background on CAN and J1939

CAN networks allowing electronic control units or sensors to exchange data at a baudrate of up to 1Mbps using a twisted two-wire cable. Transmissions over the two differential lines, CAN-H and CAN-L, are encoded using dominant (logical '0') and recessive (logical '1') bits and organized into frames with a specific bit structure. Voltage levels for transmission and reception lines between the microcontroller and the CAN transceiver, i.e. CAN-TX and CAN-RX, are nominally 5V TTL for a recessive bit and 0V TTL for a dominant bit. The voltage level is around 2.5V for both differential lines, i.e. CAN-H and CAN-L, when the bus is idle or a recessive bit is transmitted and around 1.5V on CAN-L with 3.5V on CAN-H when a dominant bit is transmitted.

Each frame starts with a dominant bit which represents the start of frame field followed by the arbitration field containing the frame identifier of 11-bit for standard frames and 29-bit for extended frames. Next is the control field which includes the number of bytes contained in the frame data field. The data field holds the actual frame content of up to 8 bytes and is followed by a CRC field used after reception for data integrity check. The last parts of the frame are the acknowledge field and the end-of-frame field. Since two or more nodes may start bit transmission at the same time, all nodes compared each own bit transmission with the actual bus value during the arbitration field. Nodes detecting a lost arbitration will have to wait for the bus to be idle again and restart the transmission. Each receiver must acknowledge the successful frame reception transmitting a dominant bit during the acknowledge field. The standard frame and extended frame structures, i.e. 11-bit ID and 29-bit ID, are shown in Figure 2. The extended frame has a higher arbitration field and structure bit count when compared to the standard frame with 20 additional bits, 18 bits for the identifier, one for the SRR (substitute remote request) bit which is always recessive and one reserved. The IDE (identifier extension) bit is recessive and part of the arbitration field for extended frames while for standard frames it is dominant and part of the control field. There are two reserved bits, i.e., r0 and r1, in the control field for extended frames and only one for the standard frame.

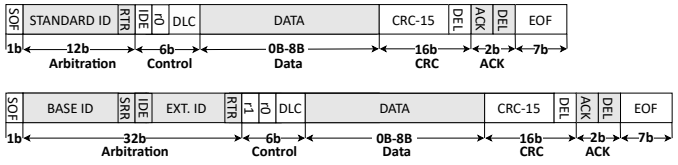


Fig. 2. Frame size and structure illustrated for standard CAN frames (top) and extended CAN frames (bottom)

J1939 specifics. Various higher layer protocols were defined over CAN such as ISO-TP and UDS for diagnostics, DeviceNet for industrial automation or SAE J1939 for commercial vehicles. All these protocols make use of the CAN physical and data link layers described earlier and define specific behavior at other OSI layers. According to the standard [16], J1939 uses 29-bit identifiers which are only found in extended frames. The 29-bit J1939 frame identifier contains a frame-priority field of 3 bits used to optimize bus traffic, a 1 bit extended data page field, required to be 0 for J1939, a data page field of 1 bit which defines the page of the Parameter Group Number (PGN), the PDU Format field of 8 bits, the PDU specific field of 8 bits and the 8-bit Source Address which identifies the frame transmitter. Source addresses are unique values and are assigned to network nodes statically or dynamically after initialization according to the J1939 standard.

B. Related works

The academic community was quick to react with various solutions for preventing attacks on in-vehicle networks reported in past years [2], [17], due to the inherent vulnerabilities of the CAN protocol. One way intruders can be prevented from transmitting malicious signals on the vehicle bus is to include authentication data inside CAN frames [18] but considering the maximum allowed payload for a CAN frame, i.e., 8 bytes, there is little space to fit both the authentication tag and actual data. The AUTOSAR standard for Secure On-board Communication (SecOC) [19] recommends truncated message authentication codes, i.e., MACs, and random freshness data bits to be included in frames with safety critical information. Another proposal is to use ID-hopping techniques [20], [21] for frame identifiers which makes it very hard or impossible for an attacker to find the correlation between the frame identifier and frame content.

Other lines of work have considered fingerprinting ECUs, removing the need of modifying the CAN frame content, by using physical characteristics, i.e., clock skews derived from cyclic frames or voltage fingerprints. Clock skews have been initially proposed to fingerprint computers [12] and well after that they been shown to be effective in fingerprinting smartphones [22]. They have also been analyzed as possible features that can be used to detect intrusions in several types of networks such as nodes in wireless sensor networks [23] and access points in wireless networks [24]. Only recently they were suggested for in-vehicle networks to identify electronic control units from the bus [25]. However, clock skews can be faked by the host and their use as fingerprints is vulnerable to cloaking attacks as shown by [15], [26]. Clock skews are still efficient for identifying legitimate ECUs which will not change their skew to evade correct classification [11]. Clock offset variations over temperature were analyzed in [27] where temperature-varied ECU fingerprinting is discussed for source identification and intrusion detection.

Fingerprinting ECUs based on voltages is a research topic that attracted much more interest in recent years. While similar approaches were used in the past for fingerprinting devices in wireless communication [28] and even wired Ethernet [29], the first work to present voltage based physical fingerprinting of CAN nodes by using basic signal processing tools is [30]. The authors in [31] used voltage profiles and ACK voltage thresholds from CAN frames to fingerprint ECUs and detect adversarial ECUs testing their approach both on experimental and real in-vehicle networks. Choi et. al [32] have used an experimental setup based on Arduino boards and real data from two cars, a Hyundai Sonata and a Kia Soul. Single frame based physical fingerprinting has been shown as feasible in [33] on extracted voltage information done with an oscilloscope and a USB data acquisition board.

A different approach is presented in [34] where voltage samples are split in 3 separate groups before being used for fingerprinting. Rising edge, falling edge and the dominant bit level are evaluated using several statistical characteristics: mean, standard deviation, variance, skewness, etc. Two vehicles were used for data collection: a Fiat 500 and a Porsche Panamera, each of them having 6 ECUs on the analyzed bus. An extension of the work was done in [35] with more details related to the outside temperature when the voltage data was collected, the number of frames and other properties for each data set. Edge based identification is studied in a recent work [36] using voltage data collected with a PicoScope 5204 and a resolution of 8 bits down to a sample rate of 2 MS/s. Kneib et. al [37] analyze the impact of voltage sampling method for a previously proposed intrusion detection methods [34], [35], [36]. They emphasize the effects of the sampling selection over the signal quality and perform an evaluation of the performance impact change due to the sampling method on a vehicle concluding that using an average of samples is the best option for the IDS. The same authors propose a low-resource constraint voltage-based intrusion detection method that can be implemented on automotive graded microcontrollers [38].

Average and standard deviation of voltage distribution were considered as features for voltage fingerprinting done in [39] on a CAN bus prototype with nodes communicating at 500 kbps. The bus contained 9 different types of ECUs along with a dedicated node which verified if the transmitters are genuine based on the specified fingerprints proving that their method reduces the false alarm rate, miss detection rate and the authentication latency if compared to the proposal from [31]. A common aspect with all these approaches is that they all fingerprint CAN nodes based on signals transmitted by individual nodes. In contrast, the work in [40] proposes fingerprinting the network layout with the ability to detect the insertion of new nodes based on time domain reflectometry. Fingerprinting ECUs using the bit time was proposed in [41] where the authors define a classification model using statistical measurements of the collected data, e.g., mean, standard deviation, etc. Similarly, the physical characteristics used in [42], i.e., voltage thresholds, rising and falling edge, open the room for detecting spoofing and bus-off attacks. Authors from [43] suggest a voltage based IDS that can be connected on the CAN bus as an independent device without influencing the behavior of the CAN network, e.g., bandwidth, and has an accuracy of more than 97% for detection of malicious voltage signals.

Xiao et. al [44] define a CAN bus authentication scheme using reinforcement learning that is based on the CAN bus physical layer features, i.e., signal voltages and arrival intervals and emphasize the framework evaluation experiments performed on a 500 kbps CAN bus with 18 legitimate ECUs and one adversarial ECU.

Additionally to the voltage fingerprinting proposals, authors from [45] evaluate voltage corruption techniques that would result in masquerade attacks not detected by voltage-based intrusion detection systems. As prevention for the voltage corruption attacks, they propose a defense method that is applied during re-training due to required fingerprint updates.

III. Vehicles and Data Collection

In this section we present the vehicles that were subject to our experiments, then we describe the tools that were used for data collection.

A. The vehicles in our experiments

For evaluating the physical characteristics of automotive control units we considered 10 different vehicles, 9 cars and 1 tractor, for which we monitored the in-vehicle CAN communication. The cars used fall in three different body configurations: hatchback (Hyundai i20, Ford Fiesta, Opel Corsa, Dacia Logan), sedan (Honda Civic) and SUV (Dacia Duster, Hyundai ix35, Ford Kuga, Ford Ecosport) with manufacturing dates between 2006 and 2021 as already shown in Table I. The diversity of cars was chosen by considering that Ford vehicles are designed in the United States of America, Hyundai and Honda originate from the Asian Market while Opel and Dacia have the design facilities in Europe. Also, the tractor employed is a contemporary machinery produced by John Deere, a famed worldwide manufacturer of agriculture equipment.

The graphical depiction of vehicles used for collecting data for our experiments and their CAN bus networks is shown in Figure 3. The topologies illustrated in the figure are derived from the identification of the ECUs based on physical characteristics as we will later detail. This does not represent the exact wiring of the ECUs inside the car of which we are unaware and which is not within the scope of our work. We are specifically interested in the number of ECUs and their fingerprints.

For some of these vehicles we were however able to identify the role of specific ECUs, as outlined in Figure 3, based on service manuals available on-line as discussed next. We identified the ECUs using the electrical wiring diagrams for the Ford Fiesta module communication networks posted at [46]. The gateway module (GWM), sync module (APIM), headlamp control module (HCM), powertrain control module (PCM), body control module (BCM) and parking aid module (PAM) are the 6 ECUs from the Fiesta OBD network. For Ford Kuga, the communication network diagram posted on [47] includes nodes communicating on the CAN bus connected to the OBD-II port, also named data link connector, i.e., DLC. In this case, we extracted data from 9 control units as follows. The keyless vehicle module (KVM), instrument cluster module (IC), powertrain control module (PCM), fuel additive system module (FUEL), ABS module (ABS), yawrate sensor (YAW), headlamp leveling module (HLM), all-wheel drive control unit (AWD) and electrohydraulic power steering module (EPS) are the Ford Kuga nodes. A Dacia Forum for UK owners [48] hosts the electrical wiring diagram for the Dacia Duster which includes the network diagrams. We used it to extract the ECU names connected on the OBD-CAN bus from the SSESP diagram. In this case there are only 3 ECUs, the ABS control unit (ABS), the injection system control unit (INJ) and the front/rear torque distribution control unit (FRTD). We determined that the John Deere tractor has 3 ECUs, the body control module (BCM), transmission control unit (TCM) and engine control module (ECM) based on the content of the frames specified in the J1939 standard.

For the rest of the vehicles, we do not know the specific role of the ECUs and we simply enumerate the ECUs in Figure 3. For Honda Civic we identified 6 distinct control units and for Opel Corsa 4 ECUs. On the Hyundai i20 we have determined that there are 7 nodes transmitting through the OBD II port. We determined that there are 6 ECUs connected on the Dacia Logan bus, 6 ECUs in the Hyundai ix35 diagnostic network and 4 ECUs in the Ford Ecosport OBD-CAN bus.

B. Tools for data collection

In order to enable the CAN data collection from the cars and tractor, we employed two devices: a CANcaseXL and a Pico Scope 5000 Series. The CANcaseXL belongs the XL Family devices produced by Vector, the most widely used networking tools provider in the automotive domain. The CAN communication networks can be interfaced through CANcaseXL using a large variety of software tools developed by Vector, e.g. CANoe, CANalyzer, CANape. We built a custom application using the XL Driver Library which facilitates easy access to several protocol specific functions, e.g., receive CAN frames with a specific baud rate, and it also supports interfacing with the CANcaseXL.

We accessed CAN data from the cars and tractor using the diagnostic port connected to the internal CAN networks having the data collection setup as shown in Figure 4 (i). The cars use communication based on the standard CAN specifications while the tractor uses SAE J1939 standard. Therefore, the J1939 OBD port has a specific shape and layout with 9 pins as described in [49]. The OBD port and the pins used to collect the data from the Ford Fiesta are shown in Figure 4 (ii) while Figure 4 (iii) depicts the specific J1939 9-PIN OBD port inside the tractor. Both pictures outline the CAN-H, CAN-L, GND pins which are of interest for collecting the skew and voltage data.

The clock skew data contains the frame identifiers and associated timestamps. In order to perform data collection for skews we built the XL Driver Library based application, configured with a baudrate of 250 kbps for the tractor and 500 kbps for the vehicles, to receive the available CAN frames through the diagnostic port. We connected the CAN cables directly to the diagnostic port CAN pins and interfaced them via a DB9 female connector to the CANcaseXL. Having the setup ready, we proceeded to log the CAN traffic over periods of 5 to 10 minutes for each vehicle while it was already turned on. Note that the CAN IDs that were collected are part of normal functioning and do not include the initialization phase, i.e. startup processes, in which some specific actions may be performed, e.g. address claiming procedures for J1939.

For voltage data collection, the probes of a 5000 Series Picoscope were connected to the CAN-H and CAN-L lines of the diagnostic port using a specific connector with accessible pins. Several files were extracted from each car and the tractor in order to capture the voltage data for all frames transmitted on the CAN bus, i.e., for each frame identifier, considering the frame identifiers extracted from the timestamp data collection step. We configured the voltage capture settings using the PicoScope 6 version 6.14.23.5207 software tool. Since the monitored lines have the maximum accepted voltage of 4.5V for CAN-H the voltage range configured on the PicoScope for both channels was of ± 5V. In order to collect an adequate amount of data required for fingerprinting transmitters, the

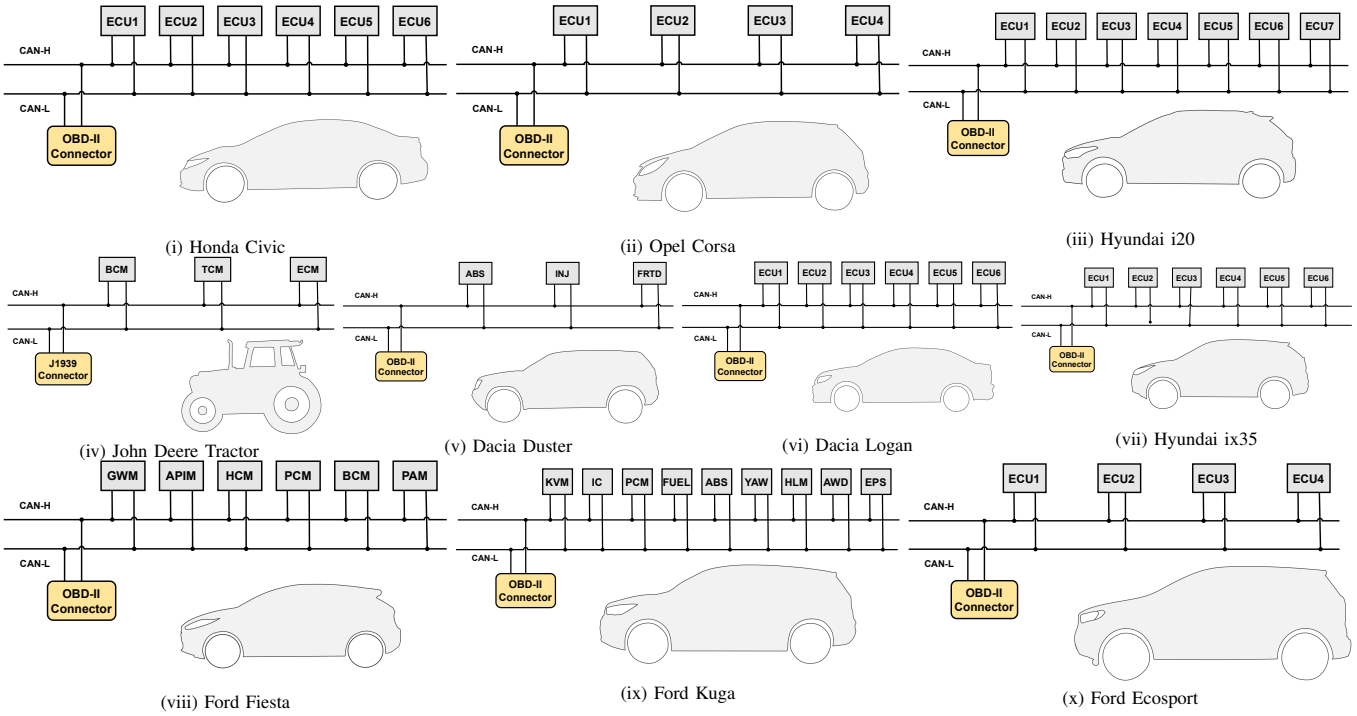
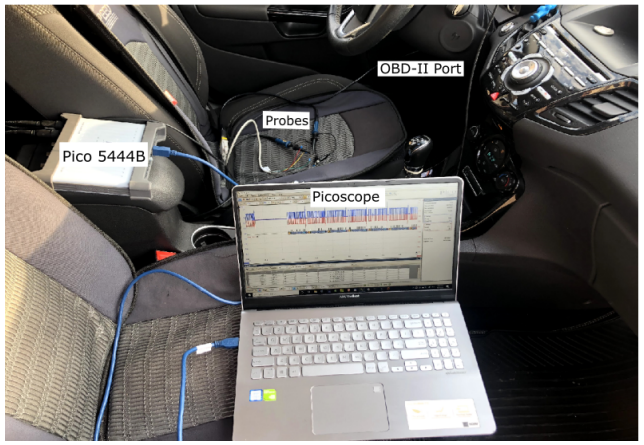


Fig. 3. ECUs communicating on the High-Speed CAN Bus accessible through the OBD-II Connector for Honda Civic (i), Opel Corsa (ii), Hyundai i20 (iii), Dacia Duster (v), Dacia Logan (vi), Hyundai ix35 (vii), Ford Fiesta (viii), Ford Kuga (ix), Ford Ecosport (x) and through the J1939 Connector for John Deere tractor (iv)



(i) data collection setup

(ii) OBD-II port (iii) J1939 port

Fig. 4. Setup for OBD-II CAN differential voltage data collection from Ford Fiesta (i), the OBD-II port from the Ford Fiesta (ii) and the J1939 diagnostic port from the John Deere tractor (iii) with specified connection points

sample rate was set to 500 MS/s, with a sample interval of 2 nanoseconds, which is maximally achievable on our Picoscope device when using two channels. Frames are transmitted by nodes from the tractor with a bit rate of 250 kbps according to SAE J1939-11, so, we were able to capture voltage data of up to 9-10 separate frames in a window of 5 milliseconds assuming a 500 microseconds average duration for one frame. For the cars, nodes exchange data using a bit rate of 500 kbps following the recommendation of SAE J2284-3 so it would take up to around 260 microseconds on average for a frame to be transmitted, allowing us to lower the capture window to 2 milliseconds. The window size and baud rate are the only differences between the tractor and the car data collection setup.

While the Picoscope tool is indeed a more expensive laboratory device, similar hardware components such as the Xilinx XC6SLX25 FPGA used in the Picoscope 5000 series or a high performance ADCs such as the 2 channel, 500 MSPS ADC08D502 from Texas Instruments are both below 100 USD. So the practical deployment of a tool with the capabilities required in our work should not be extremely expensive.

IV. Framework for Analysis

In this section we present the theoretical framework for the two fingerprinting methodologies: clock skews and voltage levels.

A. Clock skews

To set up a theoretical framework, we rely on the well established formalism from [50]. The clock of the system is defined as a piecewise continuous function $\mathbb{C} : \mathbb{R} \rightarrow \mathbb{R}$ that is

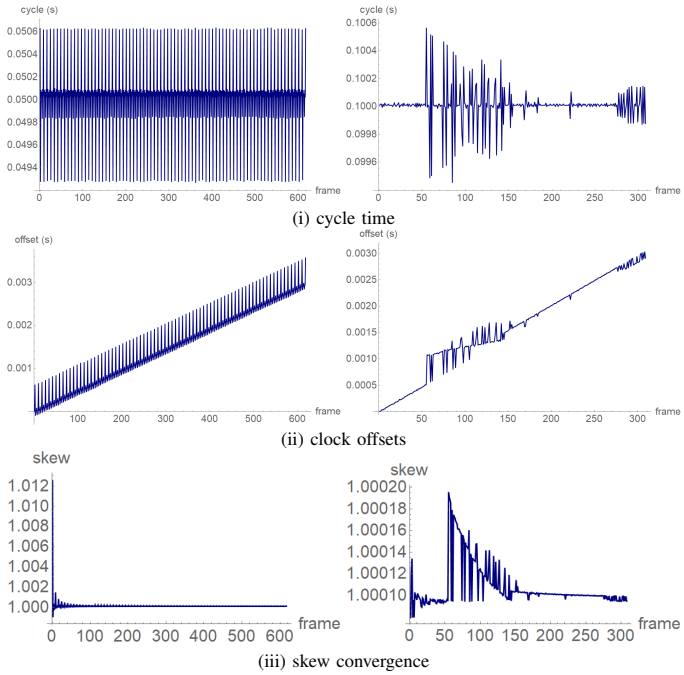


Fig. 5. Cycle time (i), clock offsets (ii) and convergence of the skew (iii) for two IDs with 50ms and 100ms cycle (left vs. right)

twice differentiable. With respect to this function, we will use the *offset* of each device clock, which is the difference between the reported time and the *true* time, e.g., $\mathbb{C}^A_{\text{offset}} = \mathbb{C}^A(t) - t$. Subsequently, the first derivative of the clock function $\mathbb{C}'(t)$ represents the clock *frequency* and its second derivative $\mathbb{C}''(t)$ is the clock *drift*. The clock *skew* between two clocks can be then defined as the difference in the first derivative, i.e., $\mathbb{C}^{A,B}_{\text{skew}} = \mathbb{C}'^A(t) - \mathbb{C}'^B(t)$. Further, the *drift* of a clock relative to another is the difference between the second derivative of the clocks, i.e., $\mathbb{C}^{A,B}_{\text{drift}} = \mathbb{C}''^A(t) - \mathbb{C}''^B(t)$.

In our practical scenario, we consider that the true clock is the frame cycle time multiplied with the frame number, i.e., $t = i \times \delta_{id}$ where δ_{id} is the cycle time of the frame carrying a specific ID. The clock skew can be estimated from cyclic frames carrying a specific ID as: $\mathbb{C}_{\text{skew}}(id) \approx \dfrac{t_j - t_i}{(j-i) \times \delta_{id}}$. Here t_i and t_j are the timestamps of the i-th and j-th frame respectively. We note that the work in [50] proposes the use of more complex linear approximation algorithms to extract the clock skews from network packets. The fact that CAN bus frames have fixed cycle time makes the previous simple formula to work reasonably well in estimating the skews if one simply takes the mean or median over the arrival time of sufficient frames. Of course, more demanding algorithms will lead to more accurate results.

To serve as a graphical example, Figure 5 shows the cycle time (i), clock offsets (ii) and the convergence of the skew (iii) computed with the above approximation for two IDs from a trace of 30 seconds. The left side of the figure is for an ID with a cycle time of 50ms and the right side for an ID with a cycle of 100ms. In our setup the *true* time is the expected arrival time of each frame based on its pre-defined cycle time and the current clock on the tool which records the data (a CANcase as earlier discussed in the data collection section). Note that while the cycle time is very distinctive for the two IDs, the slope of the clock offset is identical, i.e., the same skew equal to 1.00009, demonstrating that these IDs originate from the same ECU. The clock offset is significantly affected between the 50-th and 150-th frame for the ID on the right side of the figure, which is likely due to its lower priority. Computing the skew in this specific portion may lead to erroneous results. Such events are rare, but they show that clock skews may occasionally be unreliable and more frames have to be considered to get a correct image. As shown in (iii), for the first ID the value of the skew converges after just a dozen frames due to its more stable arrival time which is likely due to its higher priority. However, for the second ID the skew is wrongly estimated to be between 1.00010 and 1.00020 during frames 50–150. This suggests that for correct computation of the skew more than 100 frames may be needed. As a partial conclusion, skews are good for fingerprinting ECUs if there are sufficient frames but not very good for detecting intrusions in real-time since the arrival time of individual packets may be deceiving.

B. Voltage features

As already stated, another approach for fingerprinting ECUs is based on unique characteristics of the physical signals generated by each node on the CAN bus. These unique features can be observed by analyzing minute variations in the voltage levels of the CAN High and CAN Low lines. There are several causes for the uniqueness of signal characteristics. First, there are minute, uncontrollable, differences in the manufacturing process of CAN transceivers which will generate unique characteristics in their signaling behavior. Second, the shape of the transmitted signal is influenced by the transmission medium itself and all the other nodes located along the transmission path [51]. In the current analysis we focus on four features that can be extracted from the voltage levels: the mean voltage, maximum voltage, bit time and plateau time for isolated dominant bits, i.e., between two recessive bits. The last two characteristics depend also on the clock of the node. We now formally define these characteristics.

We use the following voltage features, which are going to be formalized next, and are already considered in related works: i) the mean voltage level in [34], [36], [52], ii) max voltage level in [32], [34], [36], [52] and iii) bit time in [41], [53], [54], [55]. In addition to these, we also note that the plateau time of the bit (not used in previous works) also provides good indications on the sender ECU, so we introduce this as an additional metric to compare it with existing metrics. Nonetheless, in the experimental analysis, by combining these four metrics the identification results are significantly improved. While machine learning techniques may give a much better separation between distinct ECUs, the experimental analysis that follows will show that these characteristics seem sufficient to distinguish between ECUs and thus they may serve as a compact fingerprint for ECUs inside the car.

By mean voltage we refer to the mean voltage during the dominant state of the bus, i.e., as the bit approaches

the nominal value of ∼2V. Other papers, have also included the rising and falling edges in computing the mean voltage, however, we only refer to the mean value as long as the bit stays in the dominant voltage range. For each frame carrying a specific ID, having the collected samples for an isolated dominant bit which is a transition from recessive to dominant and back, i.e., $s_1, s_2, ..., s_\ell$, we define the mean and maximum value based on a fixed range τ as follows:

$$\mathbb{V}_{\text{mean}}(id) = \text{mean}\left\{ s_i : i = \ell/2 - \tau..\ell/2 + \tau \right\}$$

$$\mathbb{V}_{\text{max}}(id) = \max\left\{ s_i : i = 1..\ell/2 - \tau \right\}$$

The value of τ has to be chosen based on the duration of a bit and the number of samples per second such that the middle portion lies on the plateau. For example, in our experiments with a 500 kbps CAN and 500 MS/s we have $\ell = 2000$ and one sample each 2ns for which we set the value of $\tau = 150$ which fitted the plateau for all bits inside all cars from the experiments.

Let indexes $\alpha \leq \ell/2, \beta > \ell/2$ and σ be the sampling time of the signal $s_1, s_2, ..., s_\ell$, then we define the bit time and plateau time as follows:

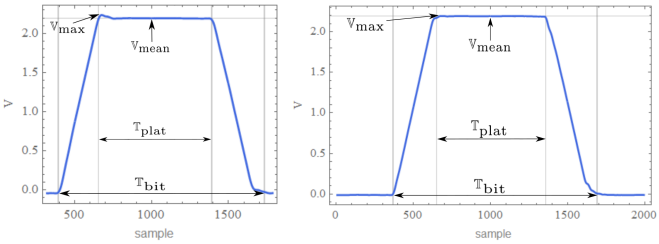
$$\mathbb{T}_{\text{bit}}(id) = \min_{\alpha,\beta} \left\{ (\beta - \alpha)\sigma : |s_\alpha| \leq \epsilon, |s_\beta| \leq \epsilon \right\}$$

$$\mathbb{T}_{\text{plat}}(id) = \max_{\alpha,\beta} \left\{ (\beta - \alpha)\sigma : |s_\alpha - \mathbb{V}_{\text{mean}}(id)| \leq \epsilon \right\}$$

For the datasets that we collected from all cars, the sampling time σ is 2ns. The threshold ϵ was selected based on empirical analysis on the intra/inter-distances at 20mV.

The previous definitions were concerned with isolated dominant bits that occurred in almost all of the IDs which we collected. If this is not the case, our approach can be easily scaled as follows. According to the CAN standard, each 5 consecutive bits of the same value are to be followed by a bit of distinct polarity (due to the CAN bit stuffing mechanism). The plateau time can be immediately extracted from multiple bits simply by dividing the entire plateau time with the number of bits (that is 5 in the worst case). Bit time can be derived from the overall time of multiple bits time by subtracting the plateau for all but 1 bit (that is the plateau time of 4 bits in the worst case). The mean and max voltage are unchanged when defined over multiple bits.

Figure 6 provides a graphical depiction of the voltage levels collected for IDs originating from distinct ECUs. The first ID is from Ford Ecosport and the second one is from Hyundai ix35. The first ID (i) shows a clear spike during the rising time with the extracted features $\mathbb{V}_{\text{mean}} = 2.195mV$, $\mathbb{V}_{\text{max}} = 2.236mV$, $\mathbb{T}_{\text{bit}} = 2.686\mu s$, $\mathbb{T}_{\text{plat}} = 1.478\mu s$. The second ID (ii) has a flatter plateau with the extracted features $\mathbb{V}_{\text{mean}} = 2.191mV$, $\mathbb{V}_{\text{max}} = 2.195mV$, $\mathbb{T}_{\text{bit}} = 2.640\mu s$, $\mathbb{T}_{\text{plat}} = 1.417\mu s$. Note that the figure presents the number of samples on the X-axis and each sampling point corresponds to 2ns, thus, the ∼1300 samples for the bit time lead to around 2.6 μs.



(i) ID 171 (ii) ID 428

Fig. 6. Collected voltage levels for IDs from distinct ECUs

As a partial conclusion, voltage levels offer much more features that can be extracted and they are also well suitable for analysis with more demanding machine learning algorithms. In this work we stay to the previously stated characteristics which are easy to collect and offer enough data to distinguish between the ECUs as shown later.

C. Distinguishability based on skews and voltage features

Having defined the skews and voltage features, we now measure the distinguishability of the fingerprints based on the intra and inter-distances between the fingerprints for each frame carrying a specific ID. Since the skews and voltages are real valued and not binary, we will use the Euclidean distance as a metric, i.e., $d(u, v) = \sqrt{(u - v)^2}$ represents the Euclidean distance of two values which can be extended to any number of features. The intra and inter-distances based on skews, i.e., $\mathcal{D}_\omega^{\text{skew}}$, voltage features, i.e., $\mathcal{D}_\omega^{\text{mean}}, \mathcal{D}_\omega^{\text{max}}$, bit time, i.e., $\mathcal{D}_\omega^{\text{bit}}$, and plateau time, i.e., $\mathcal{D}_\omega^{\text{plat}}$, where $\omega \in \{\text{inter}, \text{intra}\}$, are defined as follows:

$$\mathcal{D}^\alpha_{\text{intra}}(i) = \left\{ d(\varphi(id'), \varphi(id'')) : \forall id', id'' \in \text{ECU}_i, id' \neq id'' \right\}$$

$$\mathcal{D}^\alpha_{\text{inter}}(i, j) = \left\{ d(\varphi(id'), \varphi(id'')) : \forall id' \in \text{ECU}_i, \forall id'' \in \text{ECU}_j \right\}$$

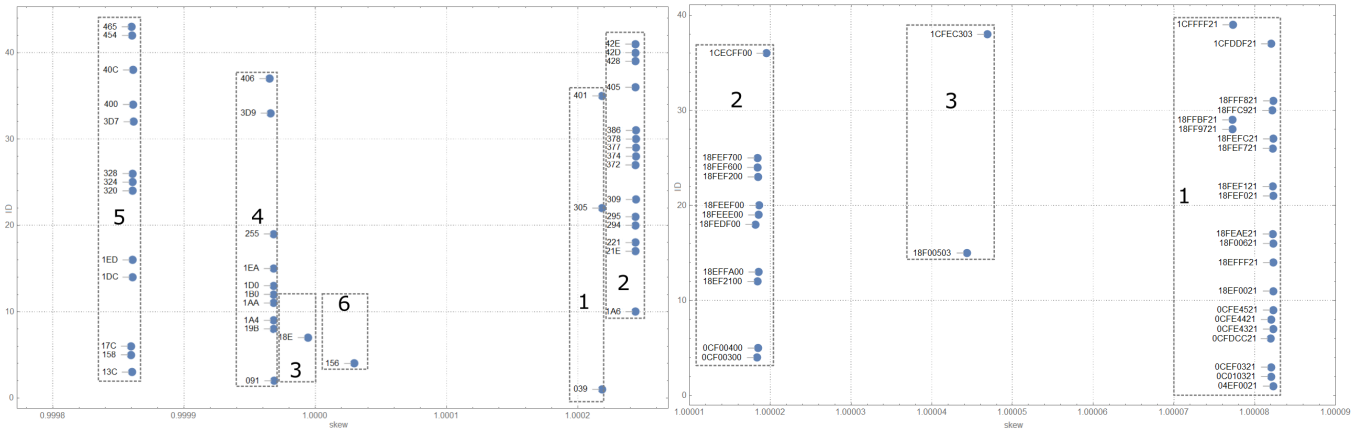
Where the pair (α, φ) runs over the five fingerprints, i.e., $(\alpha, \varphi) \in \{(\text{skew}, \mathbb{C}_{\text{skew}}), (\text{mean}, \mathbb{V}_{\text{mean}}), (\text{max}, \mathbb{V}_{\text{max}}), (\text{tbit}, \mathbb{T}_{\text{bit}}), (\text{tplat}, \mathbb{T}_{\text{plat}})\}$ and $i, j = 1..n$ run over all the n ECUs. That is, the intra-distances account for the distances between distinct IDs originating from the same ECU while the inter-distances account for the distance between IDs sent by distinct ECUs.

V. Experimental results

This section presents experimental results based on the previous methodologies and data collected inside the cars and the commercial vehicle.

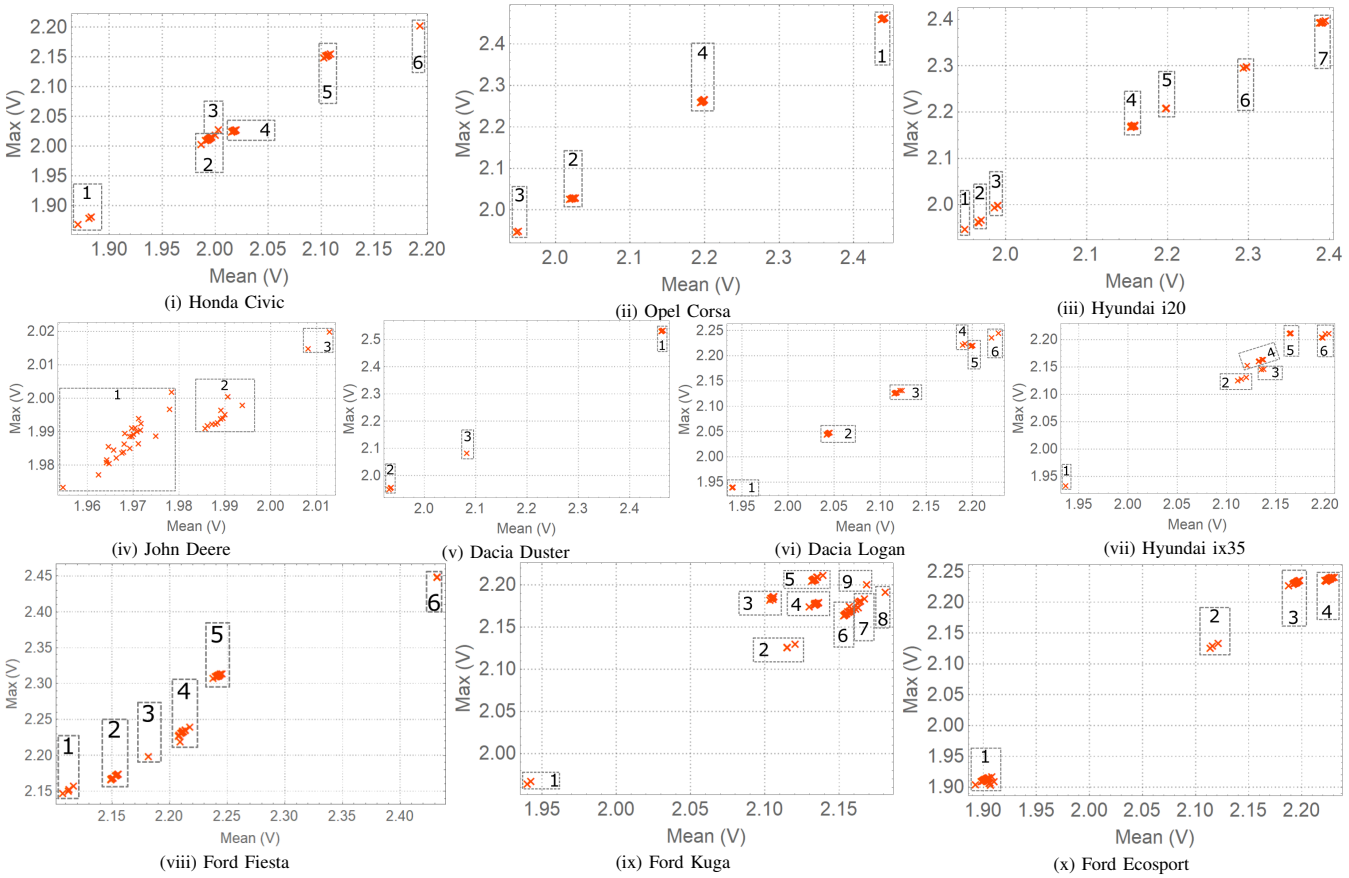
A. ECU separation based on clock skews and voltage features

We now discuss the separation between ECUs in each vehicle based on clock skews and voltage features. The results that follow indicate that neither skews, nor single voltage features like mean and max voltage, are sufficient for a fine



(i) Honda Civic (ii) John Deere

Fig. 7. Separation for ECUs in the Honda Civic (i) and John Deere (ii) based on skews



(i) Honda Civic (ii) Opel Corsa (iii) Hyundai i20

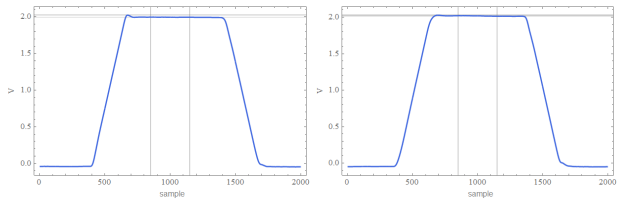
(iv) John Deere (v) Dacia Duster (vi) Dacia Logan (vii) Hyundai ix35

(viii) Ford Fiesta (ix) Ford Kuga (x) Ford Ecosport

Fig. 8. Mean-max voltage separation in the Mean-max voltage separation in the Honda Civic (i), John Deere (ii), Dacia Logan (iii), Ford Kuga (iv), Hyundai i20 (v), Dacia Duster (vi), Opel Corsa (vii), Ford Fiesta (viii), Hyundai ix35 (ix) and Ford Ecosport (x)

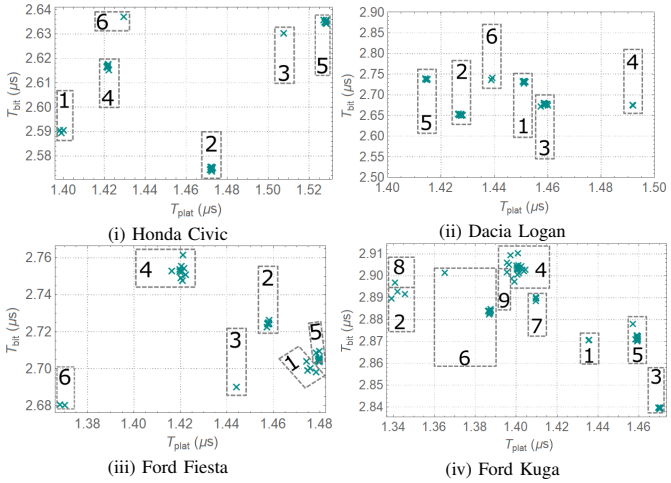
grain separation. A specific issue with skews is that frames (IDs) coming from the same ECU may have distinct skews if they are replayed by a gateway ECU. For voltage levels, the mean and max voltage may at times be very close even for distinct ECUs. These situations are outlined next in the discussions according to the case.

1) Honda Civic: we determined that there are 6 ECUs and 43 IDs on the network. The Honda Civic provides a crisp separation between 6 ECUs. Figure 7 (i) shows the separation of the ECUs based on skews suggesting 6 ECUs right from the beginning. Regarding voltages, Figure 8 (i) shows the separation of the ECUs based on the mean-max voltage features. While a separation between 5 ECUs is clear, the ID from ECU_3 is quite close to the IDs from ECU_2 and ECU_4 because of similar mean and max voltage features. However, the skew suggests this is a distinct ECU and it is indeed so as the voltage patterns from Figure 9 (i) and (ii) prove that ID 18E from ECU_3 has a distinct pattern compared with ID 091 that is originating from ECU_4. Figure 10 which provides separation based on the plateau and bit time



(i) ID 18E (ii) ID 091

Fig. 9. A zero bit from ID 18E (i) and a zero bit from ID 091 (ii) for Honda Civic



(i) Honda Civic (ii) Dacia Logan

(iii) Ford Fiesta (iv) Ford Kuga

Fig. 10. Bit-plateau time separation in the Honda Civic (i), Dacia Logan (ii), Ford Fiesta (iii) and Ford Kuga (iv)

(i) confirms that there are 6 clusters corresponding to 6 ECUs.

2) Opel Corsa: we determined that there are 4 ECUs and 29 IDs. The separation between the ECUs was one of the easiest from all cars. This was not only due to the small number of ECUs but also due to the very stable ID arrival timings and voltage features. Figure 8 (ii) shows how the IDs separate along the mean-max voltages and leaves little doubts on the separation.

3) Hyundai i20: we determined that there are 7 ECUs and 40 IDs on the network. The separation between the 7 ECUs was effortless with no overlaps in terms of skews and voltages similar to the case of the Honda Civic. We depict only the mean-max voltage clustering of the IDs against the 6 ECUs in Figure 8 (iii). Further details are given in the next section related to the inter and intra-distances.

4) John Deere: we determined 3 ECUs and 33 IDs to be present on the network. As stated, the address oriented nature of J1939 upper layer specifications provide excellent information for separating between ECUs as the sender addresses are already known. The resulting classification based on skews is presented in Figure 7 (ii) which clearly shows the IDs clustering around three physical ECUs. This supports the findings from our analysis based on the J1939 specifications which, according to sender addresses indicates the presence of 3 ECUs. We have seen that the clock skews determined for 3 IDs out of the 21 IDs sent by the BCM, i.e., ECU_1, are slightly different from the rest. However, the mean-max voltage level separation from Figure 8 (iv) confirms that these originate from the same ECU but since the skew is slightly different for those IDs, we suspect that a 4th ECU may be present behind the BCM which acts like a gateway.

5) Dacia Duster: has 3 ECUs and 12 IDs. This was the simplest topology from our cars (only 3 ECUs similarly to the John Deere tractor) and with the smallest number of IDs. Figure 8 (v) shows how the IDs separate along the mean-max voltages.

6) Dacia Logan: we determined 6 ECUs and 46 IDs. We expected a similar simplicity to the Dacia Duster but this was not the case. The fingerprint lead to the conclusion that this model has 6 ECUs. Interestingly, this car has the ECUs with the closest skews, with ECU_2 and ECU_3 differing with only 1ppm, i.e., 0.999973 vs. 0.999974. The voltages confirm that these ECUs are distinct with a large difference in the mean voltage levels of about 70mV. Figure 8 (vi) shows how the IDs separate along the mean-max voltages. Figure 10 (ii) shows the graphical separation based on the bit and plateau timings which confirms the presence of 6 distinct ECUs.

7) Hyundai ix35: we determined 6 ECUs and 26 IDs. The number of IDs is somewhat low for 6 ECUs which made us suspect that there may be a gateway behind the OBD port which filters and re-transmits incoming IDs from internal buses. The voltage features shown in Figure 8 (vii) suggest however the presence of 6 distinct ECUs.

8) Ford Fiesta: we determined 6 ECUs and 46 IDs. The Ford Fiesta creates a few problems when attempting to separate between some of the IDs using skews. The main problem is that the timing of the IDs is not very stable. By taking different portions of the trace the computed skew was slightly different. Figure 11 gives such an example. ID 073 (i) has a constant arrival time resulting in the same skew regardless of the portion of the trace. For IDs 364 (ii) and 360 (iii) there is a significant variation in the cycle time and the offsets change during the second half of the plots. This means that the skew that can be computed from the first 3000 frames is distinct from the one computed from the first 6000 frames. In this case the skew provides an unreliable separation between the ECUs. Fortunately, in this case, the mean-max ECU voltage separation works well as suggested in Figure 8 (viii). Figure 10 (iii) shows the bit and plateau time separation with 6 clusters corresponding to the 6 distinct ECUs.

9) Ford Kuga: we determined 9 ECUs and 70 IDs. We identified IDs that originate from the same ECU but have very distinctive skews again suggesting the presence of a gateway. The upper part of Figure 12 shows IDs 1D0 and 208, the first having a cycle time of 20ms and the second a cycle time of 25ms. This results in a skew of 0.999956 for the first and 1.002180 for the latter. We again suspect that the second one, i.e. ID 208, is redirected from another bus as the inter-arrival time, i.e., the cycle plot, shows significantly more noise. However, the mean-max voltage separation from Figure 8 (ix) allows us to identify separate clusters corresponding to 9 ECUs. Figure 10 (iv) contains the 9 groups of bit and plateau time which leads to the same 9 ECUs identified by the other two voltage features.

10) Ford Ecosport: we determined 4 ECUs and 87 IDs on the network. We noticed similar problems when computing the skews on Ford Ecosport as we had for Ford Kuga. The

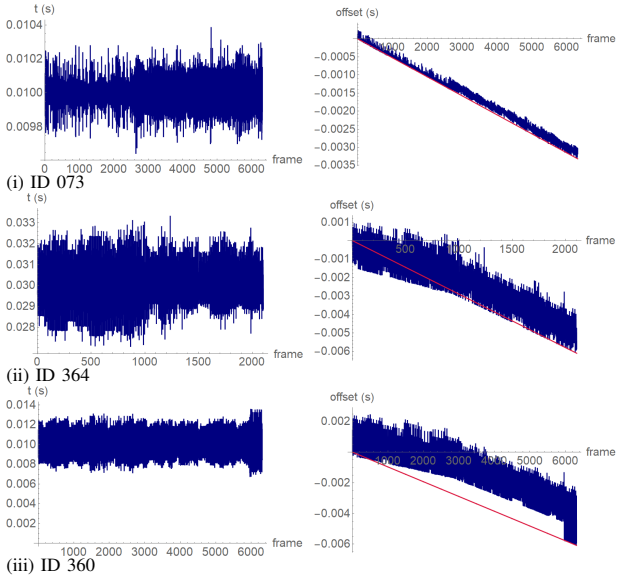


Fig. 11. Cycle time (left) and offset (right) for 3 IDs in the Ford Fiesta

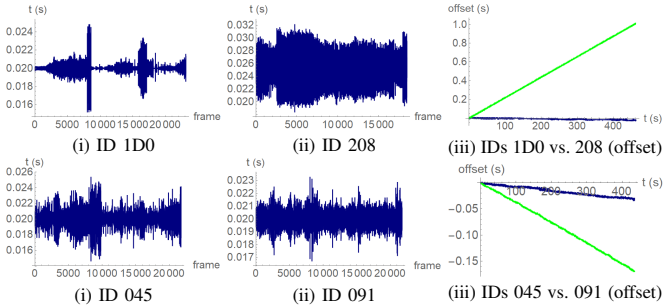


Fig. 12. Cycle time and offsets for IDs 1D0 and 208 in Ford Kuga and IDs 045 and 091 in Ford Ecosport

Ecosport shows distinct skews for IDs coming from the same ECUs. The lower part of Figure 12 shows such differences for IDs 091 having a skew of 0.999610 and ID 045 having a skew of 0.999929 both having a cycle time of 20ms although they both originate from ECU_4. Due to the clear differences in the resulted skews we suspect that ECU_4 acts as a gateway and consequently some of the IDs come with a distinct skew. Similar problems with the skews occur at ECU_1 while all the IDs from ECU_2 and ECU3 separate well from the rest. Voltage separation is almost perfect between the 4 ECUs in the Ford Ecosport as it can be seen in Figure 8 (x).

B. Overview of intra and inter-distances

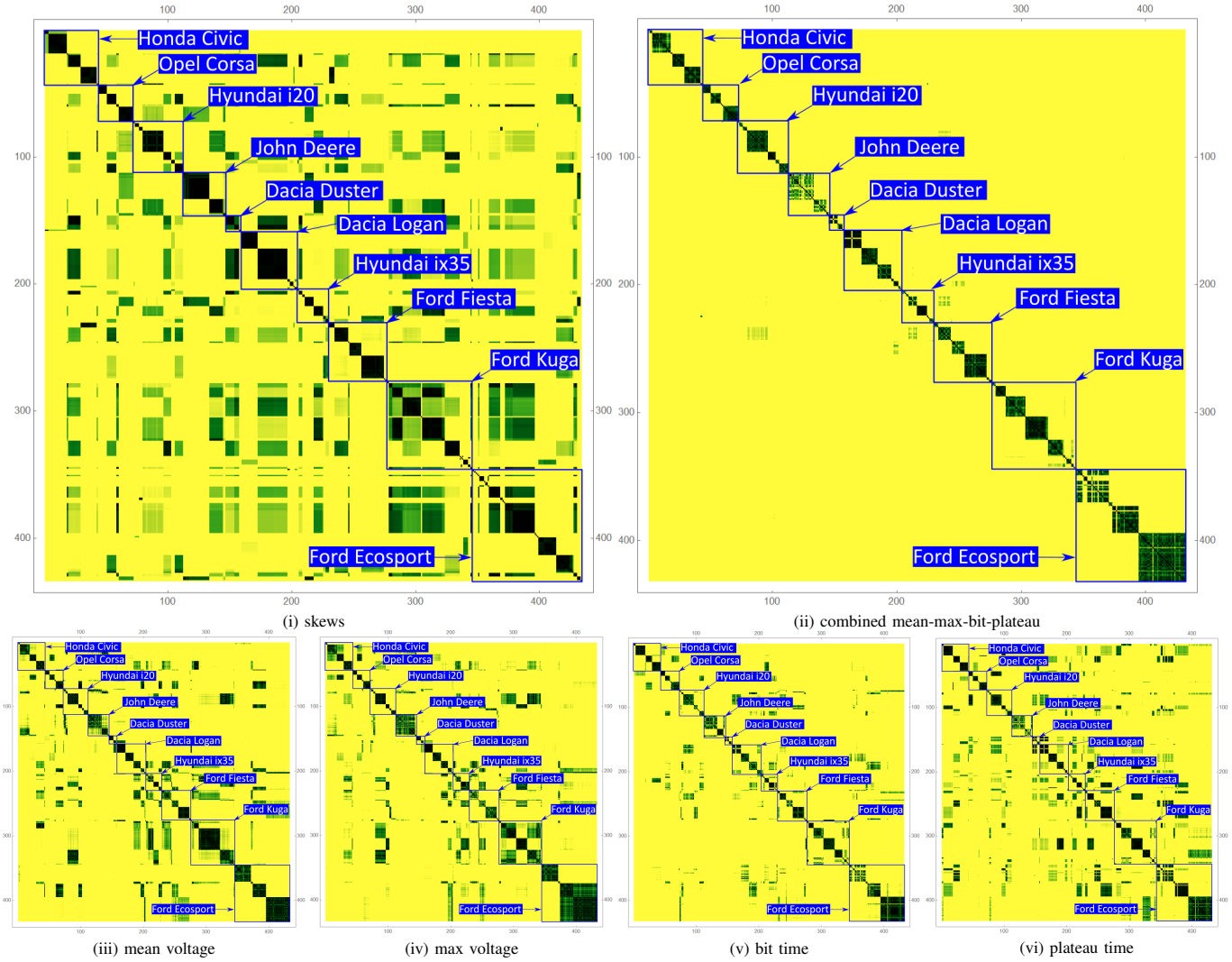
We now summarize the results on intra and inter-distances. Figure 13 shows the skews and voltage inter and intra-distances plotted as heatmaps. In the plots, we use a threshold of 10ppms for the skews, 25mV for the mean, max voltage separation and 10ns for the bit time and plateau time. The inter-distances are much cleaner for ECUs inside the same vehicles but there are far more collision in the inter-distances between ECUs in distinct cars. We discuss these in detail in what follows.

Skews provide a good separation for ECUs in the same vehicle as shown in Figure 13 (i). There are some exceptions considering the high variations in the skews extracted from the Ford Kuga and Ford Ecosport, i.e., there are more clusters based on skews compared to the number of ECUs that were separated through voltage features. Inter-distances are reasonable but insuficient as several overlaps exists, specifically for Dacia Duster, John Deere, Hyundai i20, Hyundai ix35 and Honda Civic. For example, there is a notable collision between ECU_1 and ECU_2 in the Dacia Duster and there are several collisions in the Ford Kuga or Ford Ecosport. Smaller collisions also occur between ECU_1 and ECU_2 in the Honda Civic or between ECU_5 and ECU_7 for Hyundai i20.

The mean and max voltage fingerprint from Figure 13 (iii), (iv) provide a crisper separation between the ECUs but there are still overlaps for intra-distances in the Ford Kuga, John Deere and Honda Civic. For Honda Civic there is an overlap between ECU_3, ECU_4 and ECU_5 as for John Deere the overlap is between ECU_1 and ECU_2. Still, there is a notable collision between ECU_4 in the Hyundai i20 and ECU_6, ECU_7 in the Ford Kuga. Smaller collisions also occur between ECU_2 in the Opel Corsa and ECU_3 in the Honda Civic or ECU_4 in the Opel Corsa and ECU_3 in the Ford Ecosport. The voltage patterns may still be distinctive and a finer grain analysis with machine learning algorithms will very likely distinguish between the ECUs. Such an analysis is out of scope for the current work but we leave our dataset public to be accessible for future works.

As expected since the bit time and plateau time depend both on voltages and clocks they yield a cleaner separation. The bit time heatmap from Figure 13 (v) is cleaner than the plateau time heatmap from Figure 13 (vi) with regards to inter-distances. Still, the bit time for several ECUs from the Honda Civic is close to the bit time for ECU_1 or ECU_4 from the Opel Corsa. Similarly, the bit times are close for some ECUs from the Hyundai i20, Dacia Logan, Hyundai ix35, Ford Fiesta and Ford Ecosport. As for inter-distances, there are negligible collisions for the bit time measured for 2 ECUs in Honda Civic, John Deere and Dacia Logan. Overlaps between the bit time for more ECUs are visible for the Hyundai i20, Hyundai ix35, Ford Fiesta, Ford Kuga and Ford Ecosport but there are no bit time overlaps for ECUs from Opel Corsa, Dacia Duster. The plateau time provides a clearer separation when it comes to ECUs from the same vehicle but there are more overlaps between ECUs from different vehicles as shown in Figure 13 (vi). The plateau time is also close for pairs of ECUs from John Deere, Dacia Logan, Ford Fiesta, Ford Kuga and the Ford Ecosport. The overlaps from inter-distance perspective are clearly visible between ECU_2 from Honda Civic and ECU_5 from Hyundai i20 or ECU_1 from Opel Corsa and ECU_4 from Hyundai i20. Same for ECU_3 from Dacia Logan, ECU_4 from Hyundai ix35, ECU_2 from Ford Fiesta and ECU_5 from Ford Kuga.

Finally, when merging multiple features, i.e., the mean, max, bit and plateau time, the overlaps almost fully disappear as Figure 13 (ii) shows. There is a minor overlap between the IDs from the Ford Fiesta and three other cars, Hyundai i20, Hyundai ix35 and Dacia Logan and some overlaps between IDs from the Ford Kuga and the Ford Ecosport ECUs, but the overlaps are minor. This suggests that at least 4 voltage



(i) skews (ii) combined mean-max-bit-plateau

(iii) mean voltage (iv) max voltage (v) bit time (vi) plateau time

Fig. 13. Heatmaps for skews (i), combined mean-max-bit-plateau voltages (ii), mean voltage (iii), max voltage (iv), bit time (v) and plateau time (vi) for the 432 IDs belonging to the 54 ECUs in the 10 vehicles from our experiments

features have to be used for a reliable separation.

C. Impact of environmental variations

We now put to test two cars from our experiments, i.e., the Honda Civic and the Ford Fiesta, against environmental variation effects on skews and voltages. For this purpose, we conducted tests by measuring the evolution of skews and voltages before and after one hour of normal car operation. It is known and well emphasized by previous research works that physical parameters will exhibit variations during car operation. However there are no details in related works on how these values actually change. A reason for which we proceed to further investigations. Briefly, our investigations show that variations are non-uniform and not necessarily increasing or decreasing as the car operates. This suggests that predictions will be hard or impossible to make. The use-case presented by us in the introduction will likely call for the same environment, e.g., an authorized garage, and thus environment changes should be minimal. Clearly, a more comprehensive study will be needed in order to determine how such variations occur over a larger time interval, e.g., one or several years. Such a large interval was out of reach for us in the current research communication. If insufficient over long time intervals, the proposed fingerprinting methodology will at least allow one to construct a quick mapping of the ECUs inside the car based on data collected from the bus, in a similar vein to the work in [11].

Figure 14 shows the evolution of the skews before and after 1 hour runtime for the Honda Civic (left) and the Ford Fiesta (right). In both cars the values are shifting to the right, meaning either that clocks on ECUs run faster or that the CANCase clock (used in the measurements) actually runs slower. Since the shifts are unequal, we suspect that the controllers are mainly responsible for the shift in skews and thus the oscillator begin to run slightly faster as the car is running (this may also be due to the battery charging as the car is running).

Figure 15 shows the shifts in mean voltage levels after 1 hour of runtime. For the Ford Fiesta (left) the voltage is shifting to the right showing higher voltages after 1 hour of runtime. For the Honda Civic however, the shifts are not

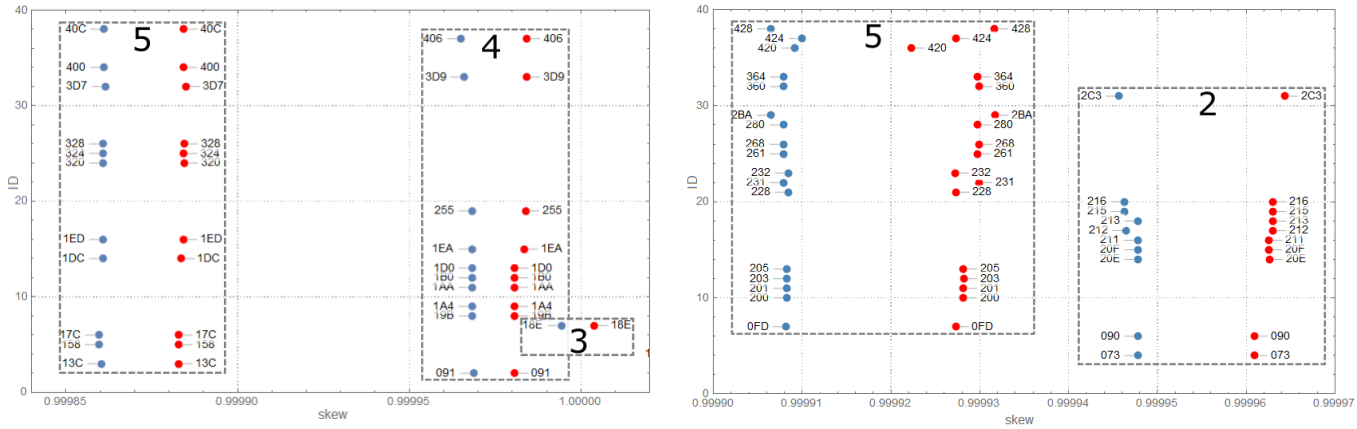


Fig. 14. Skew variations in the Honda Civic (left) and Ford Fiesta (right) from a cold start (blue) to 60 minutes driving (red)

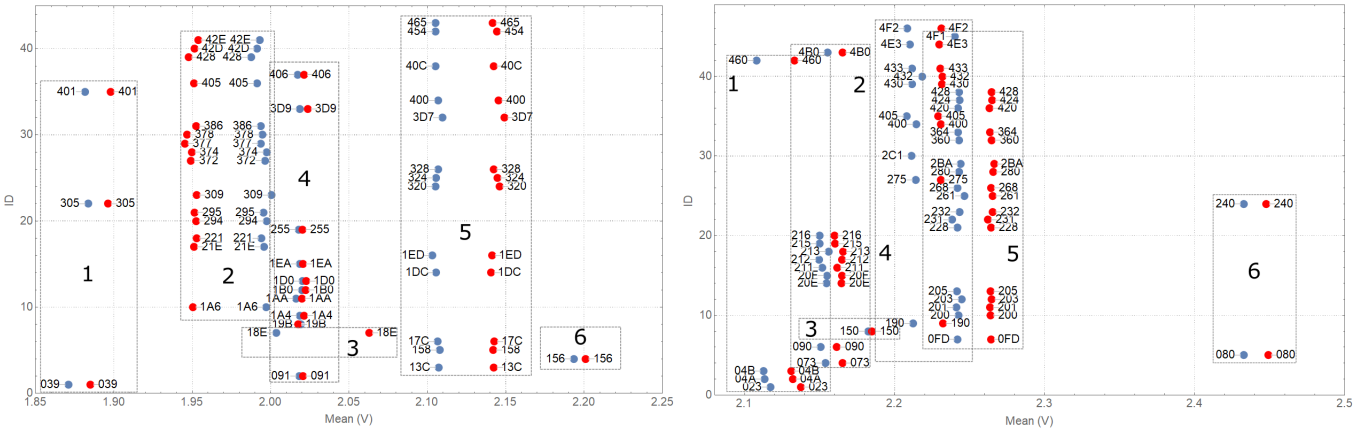


Fig. 15. Voltage variations in the Honda Civic (left) and Ford Fiesta (right) from a cold start (blue) to 60 minutes driving (red)

always positive, notably, ECU_6 exhibits negative variations. The variations are in the 10-20mV range for the Ford Fiesta but for the Honda Civic they are much higher commonly around 40mV and topping for ECU_3 at 59mV. This is generally above our separation threshold of 20mV.

The increase in both clock speeds and voltage levels makes us suspect that as the car runs the battery charges and an increased voltage supply results in faster clocks and higher voltages on the bus. This rule however does not apply to one ECU in the Honda Civic which has a decreased voltage. These variations, due to environmental changes, indicate that physical fingerprints need to be continuously updated to avoid misclassification. This has been already pointed out by previous works, but no details were given regarding these variations. Since our investigations show that these variations are not necessarily monotonic, predicting them seems infeasible in the general case. Indeed, in the case of intrusion detection systems re-training and fingerprint updates proved to be necessary as already suggested in works like [31], [34].

VI. Conclusion

Clock skews are much easier to collect, but various abnormalities due to bus loads or computational load on the host controller make skews a much less accurate and less stable fingerprinting mechanism. They also require a significant amount of frames for correct estimation (dozens to hundreds) and deviations are possible for legitimate frames which make clock skews a poor indicator when a small number of samples is available. On the other hand, clock skews may work to identify ECUs behind gateways which cannot be separated based on voltages. Voltage features are more difficult to collect, in our analysis we used a digital oscilloscope while regular in-vehicle controllers usually don't have this kind of signal acquisition capabilities. But they offer a much more accurate classification even based on single bits. Regarding voltage features, it seems that bit time and plateau time are a better feature to classify senders, however they are insufficient in a large pool of ECUs. In our analysis, grouping four features, i.e., mean voltage, max voltage, bit time and plateau time, seemed to yield a far better classification. Since our dataset is publicly available, we welcome future research works to try various other statistical tests or more demanding machine learning classifiers on the collected data.

References

[1] K. Koscher, A. Czeskis, F. Roesner, S. Patel, T. Kohno, S. Checkoway, D. McCoy, B. Kantor, D. Anderson, H. Shacham *et al.*, "Experimental security analysis of a modern automobile," in *IEEE Symposium on Security and Privacy (SP)*. IEEE, 2010, pp. 447–462.

[2] S. Checkoway, D. McCoy, B. Kantor, D. Anderson, H. Shacham, S. Savage, K. Koscher, A. Czeskis, F. Roesner, T. Kohno *et al.*, "Comprehensive experimental analyses of automotive attack surfaces." in *USENIX Security Symposium*. San Francisco, 2011.

[3] C. Miller and C. Valasek, "A survey of remote automotive attack surfaces," *Black Hat USA*, 2014.

[4] J. Van den Herrewegen and F. D. Garcia, "Beneath the Bonnet: A breakdown of diagnostic security," in *European Symposium on Research in Computer Security*. Springer, 2018, pp. 305–324.

[5] S. Nie, L. Liu, and Y. Du, "Free-fall: Hacking Tesla from wireless to CAN bus," *Briefing, Black Hat USA*, vol. 25, pp. 1–16, 2017.

[6] S. Nie, L. Liu, Y. Du, and W. Zhang, "Over-the-air: How we remotely compromised the gateway, BCM, and autopilot ECUs of Tesla cars," *Briefing, Black Hat USA*, 2018.

[7] F. Koeune and F.-X. Standaert, "A tutorial on physical security and side-channel attacks," *Foundations of Security Analysis and Design III*, pp. 78–108, 2005.

[8] M. D. Hamilton, M. Tunstall, E. M. Popovici, and W. P. Marnane, "Side channel analysis of an automotive microprocessor," in *IET Irish Signals and Systems Conference (ISSC)*. IET, 2008, pp. 4–9.

[9] M. Salfer and C. Eckert, "Attack surface and vulnerability assessment of automotive Electronic Control Units," in *12th IEEE International Joint Conference on e-Business and Telecommunications (ICETE)*, vol. 4. IEEE, 2015, pp. 317–326.

[10] T. Brennich and M. Moser, "Putting Automotive Security to the Test," *ATZelectronics worldwide*, vol. 15, no. 1, pp. 46–51, 2020.

[11] S. Kulandaivel, T. Goyal, A. K. Agrawal, and V. Sekar, "Canvas: Fast and inexpensive automotive network mapping," in *28th USENIX Security Symposium (USENIX Security 19)*, 2019, pp. 389–405.

[12] T. Kohno, A. Broido, and K. C. Claffy, "Remote physical device fingerprinting," *IEEE Transactions on Dependable and Secure Computing*, vol. 2, no. 2, pp. 93–108, 2005.

[13] B. Groza, L. Popa, and P.-S. Murvay, "CANTO-Covert AutheNtication with Timing channels over Optimized traffic flows for CAN," *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 601–616, 2020.

[14] X. Ying, G. Bernieri, M. Conti, and R. Poovendran, "TACAN: Transmitter authentication through covert channels in controller area networks," in *Proceedings of the 10th ACM/IEEE International Conference on Cyber-Physical Systems*, 2019, pp. 23–34.

[15] S. U. Sagong, X. Ying, A. Clark, L. Bushnell, and R. Poovendran, "Cloaking the clock: emulating clock skew in controller area networks," in *Proceedings of the 9th ACM/IEEE International Conference on Cyber-Physical Systems*. IEEE Press, 2018, pp. 32–42.

[16] SAE, "J1939-21 – Data Link Layer," SAE International, Standard, October 2018.

[17] T. Hoppe, S. Kiltz, and J. Dittmann, "Security threats to automotive CAN networks-Practical examples and selected short-term countermeasures," *Reliability Engineering & System Safety*, vol. 96, no. 1, pp. 11–25, 2011.

[18] A. Van Herrewege, D. Singelee, and I. Verbauwhede, "CANAuth-a simple, backward compatible broadcast authentication protocol for CAN bus," in *ECRYPT Workshop on Lightweight Cryptography*, vol. 2011, 2011, p. 20.

[19] *Spec. of Secure Onboard Communication*, AUTOSAR, 11 2020, r20-11.

[20] K. Han, A. Weimerskirch, and K. G. Shin, "A practical solution to achieve real-time performance in the automotive network by randomizing frame identifier," *Proc. Eur. Embedded Secur. Cars (ESCAR)*, pp. 13–29, 2015.

[21] B. Groza, L. Popa, and P.-S. Murvay, "Highly Efficient Authentication for CAN by Identifier Reallocation With Ordered CMACs," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 6, pp. 6129–6140, 2020.

[22] M. Cristea and B. Groza, "Fingerprinting Smartphones Remotely via ICMP Timestamps," *IEEE Communications Letters*, vol. 17, no. 6, pp. 1081–1083, 2013.

[23] D.-J. Huang, W.-C. Teng, C.-Y. Wang, H.-Y. Huang, and J. M. Hellerstein, "Clock skew based node identification in wireless sensor networks," in *IEEE GLOBECOM Global Telecommunications Conference*. IEEE, 2008, pp. 1–5.

[24] C. Arackaparambil, S. Bratus, A. Shubina, and D. Kotz, "On the reliability of wireless fingerprinting using clock skews," in *Proceedings of the third ACM conference on Wireless network security*, 2010, pp. 169–174.

[25] K.-T. Cho and K. G. Shin, "Fingerprinting electronic control units for vehicle intrusion detection," in *25th USENIX Security Symposium (USENIX Security 16)*, 2016, pp. 911–927.

[26] X. Ying, S. U. Sagong, A. Clark, L. Bushnell, and R. Poovendran, "Shape of the Cloak: Formal Analysis of Clock Skew-Based Intrusion Detection System in Controller Area Networks," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 9, pp. 2300–2314, 2019.

[27] M. Tian, R. Jiang, C. Xing, H. Qu, Q. Lu, and X. Zhou, "Exploiting temperature-varied ECU fingerprints for source identification in in-vehicle network intrusion detection," in *38th IEEE International Performance Computing and Communications Conference (IPCCC)*. IEEE, 2019, pp. 1–8.

[28] Q. Xu, R. Zheng, W. Saad, and Z. Han, "Device fingerprinting in wireless networks: Challenges and opportunities," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 1, pp. 94–104, 2015.

[29] R. M. Gerdes, M. Mina, S. F. Russell, and T. E. Daniels, "Physical-layer identification of wired ethernet devices," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 4, pp. 1339–1353, 2012.

[30] P.-S. Murvay and B. Groza, "Source identification using signal characteristics in controller area networks," *IEEE Signal Processing Letters*, vol. 21, no. 4, pp. 395–399, 2014.

[31] K.-T. Cho and K. G. Shin, "Viden: Attacker identification on in-vehicle networks," in *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security*, 2017, pp. 1109–1123.

[32] W. Choi, K. Joo, H. J. Jo, M. C. Park, and D. H. Lee, "VoltageIDS: Low-Level Communication Characteristics for Automotive Intrusion Detection System," *IEEE Transactions on Information Forensics and Security*, 2018.

[33] M. Foruhandeh, Y. Man, R. Gerdes, M. Li, and T. Chantem, "SIMPLE: Single-frame based physical layer identification for intrusion detection and prevention on in-vehicle networks," in *Proceedings of the 35th Annual Computer Security Applications Conference*, 2019, pp. 229–244.

[34] M. Kneib and C. Huth, "Scission: Signal characteristic-based sender identification and intrusion detection in automotive networks," in *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security*, 2018, pp. 787–800.

[35] M. Kneib, O. Schell, and C. Huth, "On the robustness of signal characteristic-based sender identification," *arXiv preprint arXiv:1911.09881*, 2019.

[36] ——, "EASI: Edge-based sender identification on resource-constrained platforms for automotive networks," in *Network and Distributed System Security Symposium (NDSS)*, 2020, pp. 1–16.

[37] M. Kneib and O. Schell, "Effects of the Sampling Technique on Sender Identification Systems for the Controller Area Network," *INFORMATIK 2020*, 2021.

[38] O. Schell and M. Kneib, "VALID: Voltage-Based Lightweight Intrusion Detection for the Controller Area Network," in *19th IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*. IEEE, 2020, pp. 225–232.

[39] T. Xu, X. Lu, L. Xiao, Y. Tang, and H. Dai, "Voltage Based Authentication for Controller Area Networks with Reinforcement Learning," in *IEEE International Conference on Communications (ICC)*. IEEE, 2019, pp. 1–5.

[40] M. Rumez, J. Dürrwang, T. Brecht, T. Steinshorn, P. Neugebauer, R. Kriesten, and E. Sax, "CAN Radar: Sensing Physical Devices in CAN Networks based on Time Domain Reflectometry," in *IEEE Vehicular Networking Conference (VNC)*, 2019, pp. 1–8.

[41] J. Zhou, P. Joshi, H. Zeng, and R. Li, "Btmonitor: Bit-time-based intrusion detection and attacker identification in controller area network," *ACM Transactions on Embedded Computing Systems (TECS)*, vol. 18, no. 6, pp. 1–23, 2019.

[42] J. Ning, J. Wang, J. Liu, and N. Kato, "Attacker identification and intrusion detection for in-vehicle networks," *IEEE Communications Letters*, vol. 23, no. 11, pp. 1927–1930, 2019.

[43] Y. Xun, Y. Zhao, and J. Liu, "VehicleEIDS: A Novel External Intrusion Detection System Based on Vehicle Voltage Signals," *IEEE Internet of Things Journal*, 2021.

[44] L. Xiao, X. Lu, T. Xu, W. Zhuang, and H. Dai, "Reinforcement Learning-Based Physical-Layer Authentication for Controller Area Networks," *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 2535–2547, 2021.

[45] R. Bhatia, V. Kumar, K. Serag, Z. B. Celik, M. Payer, and D. Xu, "Evading Voltage-Based Intrusion Detection on Automotive CAN," in *NDSS*, 01 2021.

[46] "Ford Fiesta Module Communications Network," https://cardiagn.com/2017-2020-ford-fiesta-ewd-module-communications-network/, 2021, accessed: 2021-07-22.

[47] "Ford Kuga Module Communications Network," http://www.fokuman.com/system_diagram-521.html, 2021, accessed: 2021-07-22.

[48] "Dacia Duster Wiring Diagrams," https://www.daciaforum.co.uk/threads/dacia-duster-electrical-wiring-diagrams.39232/, 2021, accessed: 2021-07-22.
[49] SAE, "J1939-13 – Off-Board Diagnostic Connector," SAE International, Standard,, March 2004.
[50] S. B. Moon, P. Skelly, and D. Towsley, "Estimation and removal of clock skew from network delay measurements," in *INFOCOM'99, Proceedings of 18-th Annual Joint Conference of the IEEE Computer and Communications Soc.*, vol. 1. IEEE, 1999, pp. 227–234.
[51] P.-S. Murvay and B. Groza, "TIDAL-CAN: Differential Timing based Intrusion Detection And Localization for Controller Area Network," *IEEE Access*, vol. 8, pp. 68 895–68 912, 2020.
[52] W. Choi, H. J. Jo, S. Woo, J. Y. Chun, J. Park, and D. H. Lee, "Identifying ECUs Using Inimitable Characteristics of Signals in Controller Area Networks," *IEEE Trans. Vehicular Technology*, vol. 67, no. 6, pp. 4757–4770, 2018.
[53] S. Ohira, A. K. Desta, T. Kitagawa, I. Arai, and F. Kazutoshi, "Divider: Delay-Time Based Sender Identification in Automotive Networks," in *44th IEEE Annual Computers, Software, and Applications Conference (COMPSAC)*. IEEE, 2020, pp. 1490–1497.
[54] S. Ohira, A. K. Desta, I. Arai, and K. Fujikawa, "PLI-TDC: Super Fine Delay-Time Based Physical-Layer Identification with Time-to-Digital Converter for In-Vehicle Networks," in *Proceedings of the ACM Asia Conference on Computer and Communications Security*, 2021, pp. 176–186.
[55] J. Zhou, G. Xie, S. Yu, and R. Li, "Clock-Based Sender Identification and Attack Detection for Automotive CAN Network," *IEEE Access*, vol. 9, pp. 2665–2679, 2020.

Lucian Popa started his PhD studies in 2018 at Politehnica University of Timisoara (UPT). He graduated his B.Sc in 2015 and his M.Sc studies in 2017 at the same university. He has a background of 4 years as a software developer and later system engineer in the automotive industry as former employee of Autoliv (2014 - 2018) and current employee of Veoneer (2018 - present). His research interests are in automotive security with focus on the security of in-vehicle buses.

Bogdan Groza is Professor at Politehnica University of Timisoara (UPT). He received his Dipl.Ing. and Ph.D. degree from UPT in 2004 and 2008 respectively. In 2016 he successfully defended his habilitation thesis having as core subject the design of cryptographic security for automotive embedded devices and networks. He has been actively involved inside UPT with the development of laboratories by Continental Automotive and Vector Informatik. Besides regular participation in national and international research projects in information security, he lead the CSEAMAN project (2015-2017) and the PRESENCE project (2018-2020), two research programs dedicated to automotive security funded by the Romanian Authority for Scientific Research and Innovation.

Camil Jichici is a PhD student at Politehnica University of Timisoara (UPT) since 2018 and works as a young researcher in the PRESENCE project. He received the Dipl.Ing. degree in 2016 and MsC. degree in 2018, both from UPT. His research interests are on the security of in-vehicle components and networks. He is also working as a software integrator in the automotive industry for Continental Corporation in Timisoara since 2014.

Pal-Stefan Murvay is Lecturer at Politehnica University of Timisoara (UPT). He graduated his B.Sc and M.Sc studies in 2008 and 2010 respectively and received his Ph.D. degree in 2014, all from UPT. He has a 10-year background as a software developer in the automotive industry. He worked as a postdoctoral researcher in the CSEAMAN project and is currently a senior researcher in the PRESENCE project. He also leads the SEVEN project related to automotive and industrial systems security. His current research interests are in the area of automotive security.

APPENDIX A - CONCRETE NUMERICAL DATA FOR THE COMPUTED SKEWS AND VOLTAGES

Due to space constraints, in this appendix we provide numerical data for the skews and voltages in four of the vehicles from our experiments (Dacia Logan, John Deere, Ford Fiesta and Honda Civic). Numerical data for the rest of the vehicles will be included as a supplemental material along with the datasets. All tables contain specific IDs grouped around each ECU. This association is based on the computed physical fingerprints (skews and voltages) to the best we could ascertain based on the methodology in this work. Since our dataset is public, we invite future research works to examine this association in more detail and possibly come with amendments.

The hexadecimal value of the frame identifier is presented the **ID** column and the cycle time of the frame in milliseconds, generally taken as a median, is shown in the **Cycle** column. The following columns contain data for the clock skew, $\mathbb{C}_{\text{skew}}$, the extracted mean voltage, $\mathbb{V}_{\text{mean}}$, and maximum voltage, $\mathbb{V}_{\text{max}}$, measured in Volts, and the identified bit time, $\mathbb{T}_{\text{bit}}$, and plateau time, $\mathbb{T}_{\text{plat}}$, measured in microseconds, from data collected while the vehicle was running, after it was started.

Data for the Dacia Logan is presented in Table III. Table IV contains the information for the John Deere tractor but omitting IDs 0CFFFF21, 18EAFF21, 18FEE500, 18FFFA21, 18FFFB21, 1CEBFF00, 18FFFF21, 1CECFF00 which are on-event frames and IDs which are used for multi-frame transmission. All omitted IDs for the John Deere tractor can be easily classified on the voltage levels.

Tables V and VI for the Honda Civic and Ford Fiesta have additional columns, Δ, which contain the deviations compared to the column before for each value obtained from data collected after 60 minutes of driving. In Table V we illustrate the cyclic IDs with the associated information from Ford Fiesta. The following IDs: 455, 720, 727, 728, 72F, 7A5, 7AD, have been omitted since they are on-event (non-cyclic). The last of them in particular, occurred only when the car was started and was not seen again afterwards. As skews cannot be computed for on-event IDs, we remove them from the analysis in order to make the plots in Figure 13 comparable between skews and voltage features. If needed, these IDs can be classified based on the voltage features alone. Table VI contains the measurement data for the 43 IDs from Honda Civic.

TABLE III
DACIA LOGAN

No.	ECU	ID	Cycle	$\mathbb{C}_{\text{skew}}$	$\mathbb{V}_{\text{mean}}$	$\mathbb{V}_{\text{max}}$	$\mathbb{T}_{\text{bit}}$	$\mathbb{T}_{\text{plat}}$
1	ECU1	500	100	0.997246	1.940	1.941	2.734	1.451
2	ECU1	1B0	20	0.999574	1.940	1.941	2.734	1.452
3	ECU1	552	100	0.999574	1.940	1.941	2.735	1.452
4	ECU1	657	100	0.999574	1.941	1.942	2.734	1.452
5	ECU1	2BC	100	0.999574	1.940	1.941	2.734	1.451
6	ECU1	69F	1000	0.999574	1.941	1.941	2.729	1.451
7	ECU1	4DE	100	0.999574	1.942	1.940	2.736	1.448
8	ECU1	55D	100	0.999574	1.940	1.940	2.735	1.451
9	ECU1	5DE	100	0.999574	1.940	1.941	2.733	1.452
10	ECU1	575	100	0.999574	1.940	1.941	2.734	1.451
11	ECU1	45C	100	0.999574	1.940	1.942	2.733	1.451
12	ECU1	5DF	100	0.999574	1.941	1.942	2.734	1.451
13	ECU1	350	100	0.999574	1.940	1.941	2.735	1.452
14	ECU1	4AC	100	0.999574	1.941	1.941	2.734	1.451
15	ECU2	217	20	0.999974	2.046	2.050	2.655	1.428
16	ECU2	2C6	20	0.999974	2.044	2.048	2.655	1.428
17	ECU2	2A9	20	0.999974	2.044	2.049	2.654	1.427
18	ECU2	18A	10	0.999974	2.045	2.050	2.655	1.428
19	ECU2	186	10	0.999974	2.044	2.048	2.654	1.428
20	ECU2	66A	100	0.999974	2.045	2.048	2.655	1.427
21	ECU2	511	100	0.999974	2.043	2.046	2.652	1.428
22	ECU2	1F6	10	0.999974	2.045	2.049	2.655	1.428
23	ECU2	5DA	100	0.999974	2.043	2.046	2.653	1.428
24	ECU2	648	100	0.999974	2.043	2.046	2.653	1.428
25	ECU2	65C	100	0.999974	2.042	2.045	2.653	1.428
26	ECU2	41A	100	0.999974	2.044	2.047	2.652	1.427
27	ECU2	41D	100	0.999974	2.046	2.049	2.657	1.427
28	ECU3	090	10	0.999973	2.118	2.128	2.679	1.459
29	ECU3	0C6	10	0.999973	2.116	2.126	2.681	1.459
30	ECU3	666	100	0.999973	2.124	2.133	2.674	1.458
31	ECU3	352	40	0.999973	2.117	2.128	2.677	1.460
32	ECU3	29C	20	0.999973	2.119	2.129	2.678	1.459
33	ECU3	12E	10	0.999973	2.117	2.128	2.680	1.459
34	ECU3	242	20	0.999973	2.116	2.127	2.680	1.460
35	ECU3	354	40	0.999973	2.122	2.133	2.678	1.459
36	ECU3	2B7	20	0.999973	2.118	2.129	2.680	1.459
37	ECU3	29A	20	0.999973	2.118	2.128	2.679	1.460
38	ECU3	5D7	100	0.999973	2.118	2.128	2.682	1.459
39	ECU4	1A0	100	1.000530	2.190	2.222	2.676	1.492
40	ECU4	62B	100	1.000530	2.192	2.225	2.677	1.492
41	ECU5	4F8	100	0.999507	2.201	2.222	2.739	1.415
42	ECU5	646	500	0.999507	2.200	2.222	2.742	1.414
43	ECU5	3B7	100	0.999507	2.199	2.220	2.738	1.415
44	ECU5	6FB	3000	0.999507	2.200	2.220	2.740	1.415
45	ECU6	564	100	1.000510	2.221	2.237	2.739	1.439
46	ECU6	653	100	1.000510	2.229	2.246	2.743	1.439

TABLE IV
JOHN DEERE

No.	ECU	ID	Cycle	$\mathbb{C}_{\text{skew}}$	$\mathbb{V}_{\text{mean}}$	$\mathbb{V}_{\text{max}}$	$\mathbb{T}_{\text{bit}}$	$\mathbb{T}_{\text{plat}}$
1	ECU1	18FFC921	250	1.000082	1.970	1.991	4.515	3.404
2	ECU1	18FEF021	100	1.000082	1.964	1.981	4.516	3.399
3	ECU1	0CFE4421	100	1.000082	1.965	1.981	4.517	3.398
4	ECU1	0CEF0321	100	1.000082	1.968	1.990	4.511	3.405
5	ECU1	0CFDCC21	1000	1.000082	1.966	1.982	4.510	3.400
6	ECU1	0C010321	50	1.000082	1.971	1.991	4.512	3.405
7	ECU1	1CFDDF21	500	1.000082	1.972	1.993	4.510	3.404
8	ECU1	0CFE4321	100	1.000082	1.970	1.989	4.511	3.403
9	ECU1	0CFE4521	100	1.000082	1.969	1.989	4.511	3.404
10	ECU1	18F00621	500	1.000082	1.963	1.977	4.517	3.395
11	ECU1	18FFF821	100	1.000082	1.968	1.984	4.511	3.401
12	ECU1	18FEFC21	1000	1.000082	1.955	1.974	4.513	3.404
13	ECU1	18EF0021	1000	1.000082	1.970	1.989	4.512	3.389
14	ECU1	18EFFF21	500	1.000082	1.971	1.994	4.512	3.406
15	ECU1	18FEF721	1000	1.000082	1.965	1.986	4.521	3.402
16	ECU1	18FEF121	100	1.000082	1.964	1.982	4.515	3.400
17	ECU1	18FEAE21	1000	1.000082	1.978	1.997	4.515	3.403
18	ECU1	04EF0021	20	1.000082	1.968	1.987	4.515	3.401
19	ECU1	18FF9721	100	1.000077	1.969	1.985	4.511	3.399
20	ECU1	18FFBF21	100	1.000077	1.978	2.002	4.512	3.408
21	ECU1	1CFFFF21	1000	1.000077	1.968	1.984	4.511	3.400
22	ECU2	0CF00300	50	1.000018	1.989	1.994	4.490	3.318
23	ECU2	18EFFA00	1000	1.000019	1.988	1.992	4.491	3.315
24	ECU2	18FEF600	500	1.000018	1.986	1.991	4.490	3.316
25	ECU2	18EF2100	100	1.000018	1.990	1.994	4.490	3.314
26	ECU2	18FEF700	1000	1.000018	1.989	1.997	4.491	3.326
27	ECU2	0CF00400	20	1.000018	1.990	1.995	4.490	3.319
28	ECU2	18FEF200	100	1.000018	1.987	1.992	4.490	3.320
29	ECU2	18FEEE00	1000	1.000019	1.986	1.992	4.491	3.327
30	ECU2	18FEEF00	500	1.000019	1.989	1.993	4.490	3.321
31	ECU2	18FEDF00	250	1.000018	1.994	1.998	4.491	3.315
32	ECU3	18F00503	100	1.000044	2.013	2.020	4.515	3.401
33	ECU3	1CFEC303	100	1.000047	2.008	2.015	4.516	3.401

TABLE V
FORD FIESTA

No.	ECU	ID	Cycle	$\mathbb{C}_{\text{skew}}$	Δ	$\mathbb{V}_{\text{mean}}$	Δ	$\mathbb{V}_{\text{max}}$	Δ	$\mathbb{T}_{\text{bit}}$	Δ	$\mathbb{T}_{\text{plat}}$	Δ
1	ECU1	023	100	0.999861	-0.000145	2.117	0.020	2.158	0.023	2.705	0.009	1.475	-0.004
2	ECU1	04A	100	0.999861	-0.000145	2.113	0.019	2.154	0.021	2.701	0.009	1.476	0.001
3	ECU1	04B	100	0.999861	-0.000145	2.113	0.018	2.152	0.020	2.699	0.007	1.475	0.003
4	ECU1	460	100	0.999862	-0.000147	2.108	0.025	2.148	0.029	2.699	0.013	1.479	-0.001
5	ECU2	073	10	0.999948	0.000013	2.154	0.011	2.173	0.011	2.725	0.028	1.458	-0.001
6	ECU2	090	10	0.999948	0.000013	2.151	0.011	2.168	0.011	2.725	0.024	1.458	0.000
7	ECU2	20E	10	0.999948	0.000015	2.155	0.010	2.173	0.010	2.725	0.028	1.458	-0.001
8	ECU2	20F	10	0.999948	0.000015	2.155	0.010	2.174	0.009	2.725	0.027	1.458	-0.001
9	ECU2	211	10	0.999948	0.000015	2.152	0.010	2.169	0.010	2.725	0.018	1.458	0.000
10	ECU2	212	100	0.999946	0.000017	2.150	0.015	2.167	0.015	2.723	0.023	1.457	0.000
11	ECU2	213	20	0.999948	0.000015	2.156	0.009	2.175	0.008	2.725	0.032	1.458	0.000
12	ECU2	215	20	0.999946	0.000017	2.150	0.010	2.167	0.010	2.725	0.015	1.458	0.000
13	ECU2	216	20	0.999946	0.000017	2.150	0.010	2.168	0.010	2.727	0.016	1.458	0.000
14	ECU2	2C3	1000	0.999946	0.000019	2.160	0.009	2.180	0.008	2.726	0.012	1.458	-0.001
15	ECU2	4B0	10	0.999948	0.000017	2.155	0.010	2.175	0.009	2.725	0.029	1.458	0.001
16	ECU3	150	25	1.000000	0.000009	2.182	0.003	2.200	0.004	2.691	0.004	1.444	0.000
17	ECU4	190	20	1.002000	-0.000119	2.212	0.020	2.234	0.017	2.753	0.017	1.421	-0.007
18	ECU4	275	100	1.001990	-0.000118	2.214	0.017	2.236	0.015	2.755	0.002	1.422	-0.005
19	ECU4	400	100	1.001990	-0.000118	2.214	0.017	2.236	0.014	2.749	0.019	1.420	-0.005
20	ECU4	405	100	1.002000	-0.000119	2.208	0.021	2.228	0.020	2.753	0.017	1.417	-0.002
21	ECU4	430	100	1.002000	-0.000119	2.212	0.020	2.234	0.017	2.756	0.016	1.421	-0.005
22	ECU4	432	100	1.002000	-0.000113	2.218	0.014	2.240	0.010	2.762	0.010	1.421	-0.006
23	ECU4	433	100	1.001990	-0.000108	2.212	0.019	2.235	0.015	2.751	0.018	1.423	-0.009
24	ECU4	4E3	30	1.002000	-0.000118	2.210	0.020	2.232	0.018	2.754	0.013	1.420	-0.005
25	ECU4	2C1	1000	1.001990	-0.000106	2.211	0.013	2.233	0.014	2.753	0.007	1.420	-0.002
26	ECU4	4F2	1000	1.001990	-0.000116	2.209	0.023	2.229	0.022	2.748	0.019	1.421	-0.007
27	ECU5	0FD	20	0.999908	0.000019	2.242	0.022	2.312	0.020	2.705	0.008	1.480	-0.002
28	ECU5	200	10	0.999908	0.000020	2.243	0.021	2.312	0.019	2.705	0.009	1.480	-0.002
29	ECU5	201	10	0.999908	0.000020	2.241	0.022	2.311	0.021	2.705	0.008	1.480	-0.002
30	ECU5	203	30	0.999908	0.000020	2.245	0.020	2.314	0.018	2.706	0.008	1.480	-0.002
31	ECU5	205	10	0.999908	0.000020	2.241	0.023	2.311	0.021	2.705	0.006	1.480	-0.002
32	ECU5	228	25	0.999908	0.000019	2.242	0.023	2.312	0.021	2.705	0.000	1.480	-0.002
33	ECU5	231	10	0.999908	0.000022	2.238	0.024	2.308	0.022	2.704	0.007	1.480	-0.002
34	ECU5	232	10	0.999908	0.000019	2.243	0.022	2.313	0.021	2.705	0.011	1.480	-0.002
35	ECU5	261	50	0.999908	0.000022	2.246	0.019	2.315	0.019	2.709	0.002	1.479	0.000
36	ECU5	268	10	0.999908	0.000022	2.242	0.022	2.311	0.020	2.705	0.007	1.480	0.002
37	ECU5	280	50	0.999908	0.000022	2.243	0.023	2.312	0.021	2.706	0.004	1.479	0.002
38	ECU5	2BA	100	0.999906	0.000025	2.244	0.022	2.313	0.020	2.706	0.001	1.480	-0.003
39	ECU5	360	10	0.999908	0.000022	2.243	0.022	2.313	0.020	2.705	0.007	1.480	-0.002
40	ECU5	364	30	0.999908	0.000022	2.242	0.021	2.312	0.020	2.705	0.010	1.480	-0.002
41	ECU5	420	100	0.999909	0.000013	2.242	0.021	2.312	0.020	2.705	0.012	1.480	-0.001
42	ECU5	424	100	0.999910	0.000017	2.243	0.021	2.313	0.020	2.706	0.003	1.480	-0.002
43	ECU5	428	100	0.999906	0.000025	2.243	0.021	2.312	0.020	2.705	0.001	1.480	-0.003
44	ECU5	4F1	1000	0.999905	0.000028	2.240	0.019	2.311	0.019	2.710	0.017	1.480	-0.001
45	ECU6	080	15	0.956060	-0.000290	2.433	0.016	2.450	0.012	2.681	-0.002	1.370	-0.013
46	ECU6	240	10	0.956059	-0.000270	2.433	0.015	2.449	0.013	2.681	-0.003	1.368	-0.007

TABLE VI
HONDA CIVIC

No.	ECU	ID	Cycle	$\mathbb{C}_{\text{skew}}$	Δ	$\mathbb{V}_{\text{mean}}$	Δ	$\mathbb{V}_{\text{max}}$	Δ	$\mathbb{T}_{\text{bit}}$	Δ	$\mathbb{T}_{\text{plat}}$	Δ
1	ECU1	039	40	1.000220	0.000007	1.871	0.014	1.870	0.014	2.591	0.011	1.400	-0.015
2	ECU1	305	100	1.000220	0.000007	1.884	0.013	1.883	0.012	2.590	0.011	1.399	-0.011
3	ECU1	401	300	1.000220	0.000007	1.882	0.016	1.881	0.016	2.591	0.010	1.399	-0.003
4	ECU2	1A6	20	1.000240	0.000005	1.997	-0.047	2.017	-0.050	2.575	0.003	1.473	-0.003
5	ECU2	21E	40	1.000240	0.000005	1.996	-0.044	2.016	-0.047	2.575	0.003	1.473	-0.003
6	ECU2	221	40	1.000240	0.000005	1.994	-0.042	2.014	-0.044	2.575	0.004	1.473	-0.003
7	ECU2	294	40	1.000240	0.000005	1.997	-0.045	2.018	-0.048	2.576	0.002	1.473	-0.003
8	ECU2	295	40	1.000240	0.000005	1.996	-0.044	2.015	-0.047	2.576	0.002	1.473	-0.004
9	ECU2	309	100	1.000240	0.000005	2.001	-0.048	2.020	-0.051	2.575	0.005	1.472	-0.003
10	ECU2	372	100	1.000240	0.000005	1.996	-0.047	2.016	-0.050	2.574	0.004	1.473	-0.003
11	ECU2	374	100	1.000240	0.000005	1.997	-0.048	2.018	-0.052	2.575	0.000	1.473	-0.002
12	ECU2	377	100	1.000240	0.000005	1.994	-0.049	2.013	-0.051	2.575	0.002	1.472	-0.003
13	ECU2	378	100	1.000240	0.000005	1.995	-0.048	2.015	-0.050	2.575	0.002	1.473	-0.003
14	ECU2	386	100	1.000240	0.000005	1.994	-0.041	2.014	-0.043	2.576	0.005	1.473	-0.003
15	ECU2	405	300	1.000240	0.000005	1.991	-0.040	2.011	-0.041	2.576	0.002	1.472	-0.002
16	ECU2	428	300	1.000240	0.000005	1.987	-0.040	2.004	-0.040	2.575	0.005	1.472	-0.003
17	ECU2	42D	300	1.000240	0.000004	1.991	-0.040	2.011	-0.042	2.575	0.004	1.472	-0.003
18	ECU2	42E	300	1.000240	0.000004	1.993	-0.040	2.012	-0.041	2.576	0.004	1.473	-0.004
19	ECU3	18E	10	0.999994	0.000010	2.003	0.059	2.029	0.058	2.631	0.010	1.508	-0.003
20	ECU4	091	10	0.999969	0.000011	2.018	0.002	2.027	0.004	2.617	0.010	1.422	0.002
21	ECU4	19B	10	0.999968	0.000012	2.019	-0.002	2.028	-0.001	2.617	0.010	1.422	0.002
22	ECU4	1A4	20	0.999968	0.000012	2.018	0.003	2.028	0.004	2.617	0.010	1.422	0.002
23	ECU4	1AA	20	0.999968	0.000012	2.016	0.004	2.026	0.004	2.617	0.011	1.422	0.002
24	ECU4	1B0	20	0.999968	0.000012	2.020	0.002	2.029	0.003	2.617	0.012	1.422	0.002
25	ECU4	1D0	20	0.999968	0.000012	2.020	0.002	2.030	0.003	2.617	0.011	1.422	0.002
26	ECU4	1EA	20	0.999968	0.000015	2.019	0.002	2.028	0.003	2.617	0.013	1.421	0.002
27	ECU4	255	40	0.999968	0.000015	2.018	0.003	2.027	0.003	2.618	0.010	1.422	0.001
28	ECU4	3D9	200	0.999966	0.000018	2.018	0.005	2.028	0.006	2.616	0.014	1.422	0.001
29	ECU4	406	300	0.999965	0.000019	2.017	0.004	2.027	0.002	2.618	0.013	1.422	-0.003
30	ECU5	13C	10	0.999860	0.000023	2.107	0.035	2.155	0.035	2.635	0.010	1.528	0.001
31	ECU5	158	10	0.999860	0.000023	2.108	0.034	2.155	0.035	2.635	0.010	1.528	0.002
32	ECU5	17C	10	0.999860	0.000023	2.107	0.036	2.154	0.037	2.636	0.010	1.528	0.001
33	ECU5	1DC	20	0.999861	0.000023	2.105	0.035	2.153	0.036	2.635	0.008	1.528	0.001
34	ECU5	1ED	20	0.999861	0.000023	2.103	0.038	2.151	0.038	2.635	0.009	1.529	0.001
35	ECU5	320	100	0.999861	0.000023	2.105	0.040	2.152	0.044	2.636	0.011	1.528	0.001
36	ECU5	324	100	0.999861	0.000024	2.105	0.039	2.153	0.036	2.635	0.008	1.528	-0.001
37	ECU5	328	100	0.999861	0.000024	2.107	0.035	2.155	0.035	2.636	0.006	1.528	0.001
38	ECU5	3D7	200	0.999862	0.000023	2.109	0.039	2.157	0.040	2.636	0.020	1.529	0.001
39	ECU5	400	300	0.999861	0.000023	2.107	0.039	2.155	0.037	2.636	0.008	1.529	-0.001
40	ECU5	40C	300	0.999861	0.000023	2.105	0.037	2.153	0.036	2.635	0.012	1.529	0.001
41	ECU5	454	300	0.999860	0.000024	2.105	0.039	2.154	0.038	2.636	0.009	1.528	0.001
42	ECU5	465	300	0.999860	0.000024	2.105	0.037	2.152	0.038	2.635	0.010	1.528	0.002
43	ECU6	156	10	1.000030	0.000017	2.194	0.007	2.204	0.005	2.637	0.016	1.430	-0.013