Silicon single-electron random number generator based on random telegraph signals at room temperature

Kouta Ibukuro,[1, a)] Fayong Liu,[1] Muhammad Khaled Husain,[1] Moïse Sotto,[1] Joseph Hillier,[1] Zuo Li,[1] Isao Tomita,[2] Yoshishige Tsuchiya,[1] Harvey Rutt,[1] and Shinichi Saito[1, b)]

[1)] *School of Electronics and Computer Science, University of Southampton. University Road, Southampton, SO17 1BJ, United Kingdom.*

[2)] *Department of Electrical and Computer Engineering, National Institute of Technology, Gifu college. 2236-2 Kamimakuwa, Motosu, Gifu, 501-0495, Japan.*

(Dated: 11 September 2020)

The need for hardware random number generators (HRNGs) that can be integrated in a Silicon (Si) complementary-metal-oxide-semiconductor (CMOS) platform has become increasingly important in the era of the Internet-of-Things (IoT). Si MOSFETs exhibiting random telegraph signals (RTSs) have been considered as such a candidate for HRNG, though its application has been hindered by RTS's variability and uncontrollable, unpredictable characteristics. In this paper, we report the generation and randomness evaluation of random numbers from RTSs in a Si single electron pump (SEP) device at room temperatures. SEP devices are known to consistently produce RTSs, due to a quantum dot electrically defined by multi-layer polycrystalline Si gates. Using RTSs observed in our devices, random numbers were extracted by a classifier supported by supervised learning, where part of data was used to train the classifier before it is applied to the rest to generate random numbers. The random numbers generated from RTSs were used as inputs for the Monte Carlo method to calculate the values of π, and the distribution was compared against the result obtained from Mersenne twister, a representative pseudo random number generator (PRNG), under the same condition. π was estimated more than 80000 times, and the distribution of the estimated values has a central value at 3.14 with a variance of 0.273, which is only twice larger than the result from PRNG. Our result paves a way to fully electronic CMOS compatible HRNG that can be integrated in a modern System-on-a-Chip in IoT devices.

I. INTRODUCTION

Password authentication is one of the most widely used protocols to protect one's private information[1–3]. This protocol accepts input from a user in a form of a password, and from this password a function called 'hashing'[4,5] (md5()[6], for example) generates a 128bit output, which is stored in the server. From this generated sequence of characters, called hash, it is virtually impossible to guess the original password. When the server is attacked, the hackers will obtain hashes, but not original passwords. Therefore, the security of the user is still protected. However, it is critically important to use a password which is new, long enough and random enough to avoid a so-called 'dictionary attack'[1–3], where a hacker tries to guess the password from common candidates, such as those on personal details (for example, birthday, surname and forename), a password used in elsewhere and simple sequences of numbers and letters (1234567890, abc), that matches the obtained hash. To generate a password with enough strength, pseudo random number generators (PRNGs) are used to suggest strong passwords. PRNGs use a date and time, for example, as a seed, and an algorithm produces random numbers as a password.

Another way of generating a strong password is to use a hardware random number generator (HRNG)[7–10]. HRNG uses the stochastic nature of certain physical phenomena or physical values, such as thermal noise of electronic devices[7], detection of single photons[8], timing of Ovonic threshold switching[9], chaotic behaviour of semiconductor lasers[10] and so forth. As HRNG does not rely on algorithm, the security of a password that is generated by HRNG is fundamentally superior to the one generated by PRNG.

Recently, theoretical[11] and experimental[12–14] studies have sought the possibility of utilising random telegraph signals (RTSs) for random number generation. RTSs are a stochastic shift of threshold voltage over time[15], and were first found in 1985 in silicon (Si) metal-oxide-semiconductor field effect transistors (MOSFETs)[16]. The shift in $V_{\rm th}$ has been considered to be caused by trapping and de-trapping of a single carrier (an electron in a case of n-type MOSFET (n-MOSFET), and a hole in a case of a p-type MOSFET (p-MOSFET)) into a charge trap in a gate-oxide[15]. As a consequence of aggressive scaling of Si technology, RTSs started to become recognised as a reliability issue in complementary MOS (CMOS) technology in the 2010s[17–23]. The merit of using RTSs as a source of randomness is primarily because of the compatibility of CMOS technology[15]. Also, it is well known that in many cases the probability to observe two discrete current states (high and low current states) can be well controlled by gate voltage[15,24–27], such that one can identify a bias condition where two current states are equally observed ('balance point')[9]. The difficulty in utilising RTSs as HRNG is that the presence and absence of RTSs cannot be controlled, and usually the probability to find a device that exhibit RTSs is less than 1%[15,28].

In this paper, we report the generation and evaluation of random numbers from RTSs in a Si single electron pump (SEP) device at room temperature. Our device is based on silicon-on-insulator (SOI) Fin-FET structure with multiple

[a)] Electronic mail: K.Ibukuro@soton.ac.uk

[b)] Present address: Center for Exploratory Research Laboratory, Research & Development Group, Hitachi, Ltd. Tokyo 185-8601, Japan; Electronic mail: S.Saito@soton.ac.uk poly-crystalline Si gates, and the formation of an electrically defined quantum dot (QD) is achieved when negative bias is applied to the two gates[25,26]. Our previous study reveals that RTSs can be caused by trapping and de-trapping of an electron in the QD, whose charge state is read by the net current flowing through the device[25,26]. We selected a different device this time, and RTSs were observed again, showing the expected characteristics such as average lifetime of about 1s and tunability of the charge state by the top gate. Moreover, we observed anomalous three-level RTSs, which can be explained by the presence of multiple energy levels in a QD[25]. The advantage of this mechanism is that we do not need to rely on fluctuant device features like oxide traps[13,14] or incident potential pocket[12], whose presence can only be determined probabilistically. With the future possibility of real-time random number generation in sight, we considered the generation of random numbers from RTSs as a classification problem of time-series data[29]. That is, the classifier was built from training data before it was applied to a new data to achieve random number generation. Finally, the quality of random numbers was evaluated by calculating values of π using Monte Carlo method with the generated random numbers as input. π was calculated more than 80000 times, whose distribution has a median at 3.14 with a deviation of 0.273. This variance was only twice larger than the distribution obtained from Mersenne twister (MT), a representative pseudo random number generator (PRNG), calculated under the same conditions. The reason for the wider variance was attributed to the misclassification in the test data.

II. DEVICE FABRICATION

Our device was fabricated on a 6-inch SOI wafer with the surface orientation being (100). The stack of the wafer was 100nm-thick SOI and 145nm-thick buried oxide (BOX) on a handle layer. After the thickness of the SOI was thinned down to 24nm, the nanowire and mesa for source and drain (S/D) were patterned by electron beam lithography (e-beam). This was followed by anisotropic wet etching using tetramethylammonium hydroxide (TMAH) as wet etchant. This resulted in the sidewall of the nanowire being atomically flat (111) crystalline surface[30–32]. The nanowire width of e-beam mask was designed to be 100nm. After the oxidation and wet etching, the width of the nanowire was further reduced to about 60nm. 17.6nm-thick silicon dioxide (SiO_2) for gate dielectric was grown on SOI at 1000°C, followed by its partial removal for S/D contact. Then, polycrystalline Si (poly-Si) was deposited across the entire wafer, and phosphorous was heavily doped using spin-on-dopant (SOD) technique aiming for $5\times10^{19}\mathrm{cm}^{-3}$. The dopants were activated by rapid-thermal-annealing (RTA) at 950°C for one minute. This layer was patterned by e-beam and inductively-coupled-plasma (ICP) etching for raised S/D as well as gates for electrically defining a QD in the channel, called left-gate (LG) and right-gate (RG). The length of L/RG ($L_{\mathrm{L/RG}}$) was 100nm. In order to allow the formation of an inversion layers in the regions not covered by the L/RG, another layer of poly-Si was deposited and patterned using the same technique used for L/RG. This gate is called top-gate (TG). TG and L/RG were electrically insulated by 9-nm thick thermal oxide. The length of the TG (L_{TG}) is defined as the separation between LG and RG, which is 150nm. Figure 1 (a) shows a schematic of cross-section of our device along the nanowire. Formation of the passivation oxide was followed by contact opening, aluminium deposition and its patterning for metal interconnect. Finally hydrogen termination was achieved by forming gas anneal at 450°C for 30 minutes[25].

Figure 1 (b) shows a scanning-electron-microscope (SEM) image of a similar device before the deposition of poly-Si for TG, in which the position of a QD is highlighted. Figure 1 (c) shows the potential profile along the nanowire when LG and RG were kept around V_{th} while TG inverted the entire channel. RTSs were consistently observed in devices from the same fabrication lot[25,26], and the physical mechanism of the RTSs were attributed to a trapping and detrapping of a single electron in the electrically defined QD. This kind of discretised signals were observed at room temperatures in devices fabricated using the similar technique, while the trapping and detrapping of a single electron was detected by a nearby charge sensor[33,34]. Therefore, the observation of RTSs in this particular device was also expected.

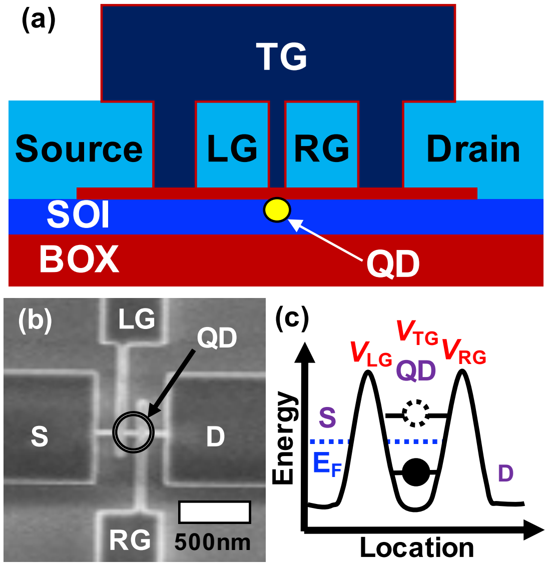


FIG. 1. Overview of the fabricated device. (a) A schematic of cross-section of device along the nanowire. (b) Scanning electron micrograph of the device. (c) Schematic of the potential profile realised in the device at room temperature when V_{LG} and V_{RG} are creating potential barriers in the channel.

III. CHARACTERISATION

A Keysight B1500 and Cascade M150 probe station were used for electrical characterisation. The detection limit of this experimental setup is 100fA. Figure 2 (a), (b) and (c) show the transfer characteristics of the device when TG, LG and RG were swept, respectively, at two different drain voltage (V_d) values, 50mV and 1V. Leakage current was absent between the channel and the gates as well as between the gates. The device exhibited acceptable behaviour as a MOSFET, with the subthreshold swing of the I_d - V_{TG} curve being 77mV/decade. The degradation in subthreshold swing in I_d - V_{RG} and I_d - V_{LG} curves was attributed to the bias condition of the measurement, where voltage on TG (V_{TG}) was set to 0V. Threshold voltage of TG (V_{th}) was 0V, and above V_{th}, current fluctuations were observed in the transfer characteristics. Similar current fluctuations were observed in a device from the same fabrication process, which has shorter gate lengths as well as narrower channel width[25,26]. This prompted to perform time domain measurements to identify the nature of the current instability in the transfer characteristics.

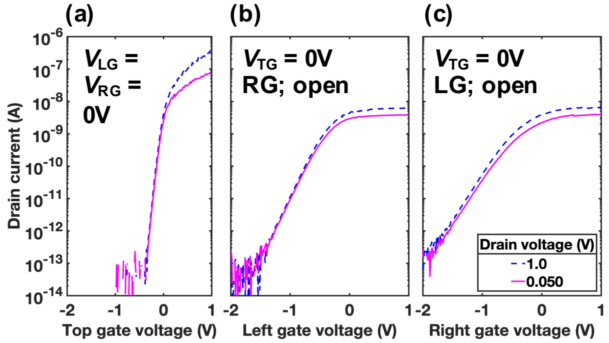


FIG. 2. Transfer characteristics of the device. (a) I_d-V_{TG} curves with V_d = 50mV and 1V. V_{LG} and V_{RG} were 0V. (b) I_d-V_{RG} curve with V_d = 50mV and 1V. V_{TG} was 0V and V_{LG} was floating. (c) I_d-V_{LG} curve with V_d = 50mV and 1V. V_{TG} was 0V and V_{RG} was floating.

Figure 3 (a) shows one example of the results of the time domain measurements, and (b) shows the histogram of the corresponding time trace, which is a probability distribution against the value of drain current (I_d). V_{TG} was set to be 0.2V, and V_{LG} and V_{RG} were -50mV in this case. The average lifetimes of $|0\rangle$, $|1\rangle$ were about 1s, while the lifetime of $|2\rangle$ was shorter than the other two current states. In Figure 3 (b), two major peaks and one minor peak can be seen. From this distribution, the probability to observe the highest current state ($|0\rangle$) and the middle state ($|1\rangle$) is almost the same, while the probability to observe the lowest current state ($|2\rangle$) is significantly lower than $|0\rangle$ and $|1\rangle$. Figure 3 (c) shows the histograms at different voltage conditions, where V_{LG} = V_{RG} were varied from -200mV (blue dash line) to 0mV (green solid line) with 50mV increments. When V_{LG} = V_{RG} = -200mV, the dominant current state was $|0\rangle$. As V_{LG} = V_{RG} increased, $|1\rangle$ started to observe more frequently, and when V_{LG} = V_{RG} = 0mV, the dominant current state was $|1\rangle$. Figure 3 (d) shows the occupancy of the current states as a function of V_{LG} = V_{RG}. The balance point was achieved at around V_{LG} = V_{RG} = -50mV. Also, as V_{LG} = V_{RG} approaches to 0mV, the probability to observe $|2\rangle$ became negligible.

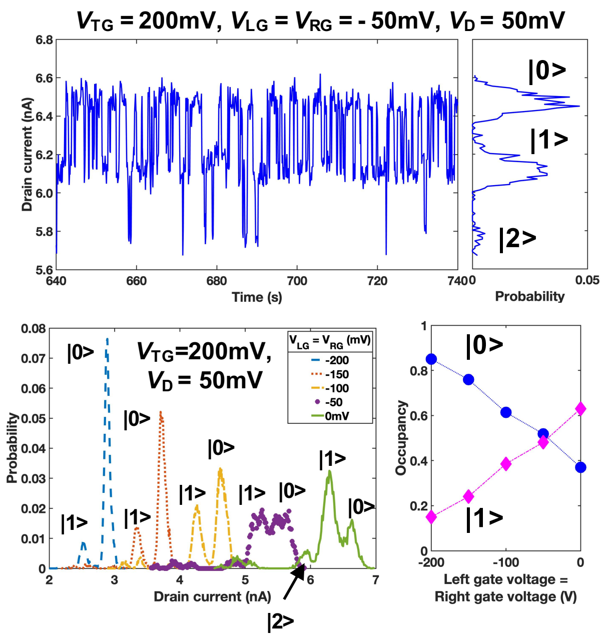


FIG. 3. Random telegraph signals observed in the device. (a) Part of a time domain measurement of the device with V_{TG} = 200mV, V_{LG} = V_{RG} = -50mV, V_d = 50mV. (b) Histogram of the corresponding time domain characteristics shown in (a). (c) Histograms of the several time domain characteristics with different V_{LG} = V_{RG} values. (d) Occupancy of the current states as a function of V_{LG} = V_{RG}.

In this device, three current states were observed, which cannot be captured by a simple physical picture describing RTSs. In the most simple case of RTSs with two current states, this can be explained by a presence of a single isolated energy level with two different charge states, occupied or empty, corresponding to the two current levels[15]. If there are n independent energy levels were present, the number of current levels should be 2^n. In order to explain the existence of three current levels, a following physical model was constructed, which is shown as schematics in Figure 4.

Four diagrams that represent potential diagrams from source to drain are displayed in Figure 4. In this model, we assume the existence of two discrete energy levels in the QD. The separation between the energy levels are expected to be in the order of 100mV from our previous result[25], justifying the observation of single electron phenomena at room temperature. When the energy barriers created by LG and RG were high (V_{LG} = V_{RG} = -200mV, for example), both energy levels are above Fermi energy of source (E_F) and the ground state is occasionally occupied by an electron and it would be released soon after (Figure 4 (a)). This occasional occupation of the level in the QD and following release caused corresponding positive shift in V_{th} followed by negative V_{th} shifts at random. As the probability for the level to be occupied is still low, the probability distribution is highly asymmetric in favor of $|0\rangle$. When the barrier is lowered and the ground state aligned with E_F, the probability to observe $|0\rangle$ and $|1\rangle$ is almost equal (Figure 4 (b)). Finally, as the barriers were further lowered, both levels can interact with the electron reservoir. This means that the ground state can still capture and release an electron (Fig-

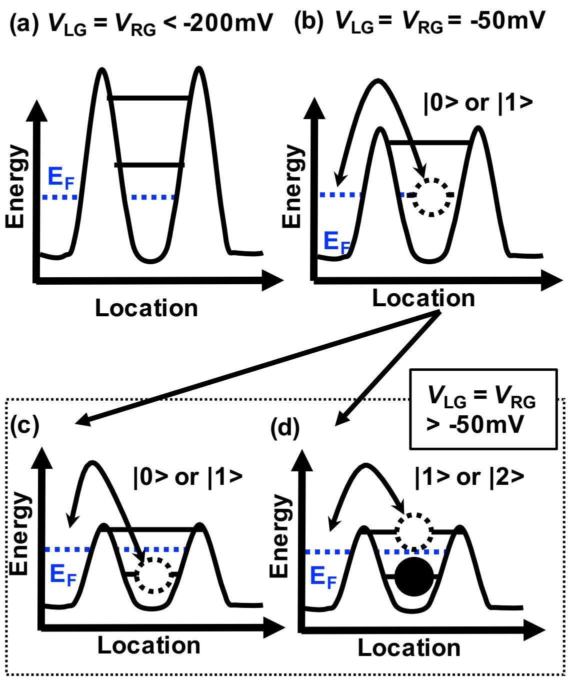

(a) $V_{LG} = V_{RG} < -200mV$
(b) $V_{LG} = V_{RG} = -50mV$
Energy
E_F
Location
|0> or |1>
$V_{LG} = V_{RG} > -50mV$
(c)
(d)
|1> or |2>

FIG. 4. Schematics of the physical model to explain the RTS shown in Figure 3. (a) The energy levels in the QD are above E_F when V_{LG} = V_{RG} are less than -200mV. (b) As the potential barriers become low (V_{LG} = V_{RG} = -150mV and -100mV), the ground state starts to be resonant with E_F. (c) The further decrease in potential barriers (V_{LG} = V_{RG} = -50mV and 0mV) brings the second energy level in resonant with E_F, while electron can be trapped and de-trapped in the ground state. (d) However, when an electron occupies the QD, another electron can occupy the excited state. The occupation of the excited state is assumed to be only allowed when the ground state is occupied.

ure 4 (c)), and the excited states can also trap and de-trap an electron (Figure 4 (d)). However, the trapping and de-trapping of an electron in an excited state can be considered to be allowed only when the ground state is occupied. Here, we exclude a possibility where the ground state is empty and the excited state is occupied, which is a reasonable assumption based on this multi-level model in a single QD. Indeed, in the time domain measurement shown in Figure 3 (a), the transition to the current state $|2\rangle$ seems mostly occurred from $|1\rangle$. Overall, this physical model explains the characteristics of the observed RTSs reasonably well.

IV. THE PROCEDURE TO EXTRACT CHARACTERISTIC PARAMETERS OF RTS FOR RANDOM NUMBER GENERATION

Following on from discussing the physical model behind the RTS observed in our device, in the next two sections the procedure of random number generation from the RTS is introduced. Data for random number generation was prepared as follows. Nine time domain measurements were taken (called Data1,2,3...9) at the fixed bias condition of V_{TG} = 300mV and V_{LG} = V_{RG} = -106mV, which still realised the potential profile depicted in Figure 4 (b). V_{TG} was increased from 200mV (as shown in Figure 3) to 300mV, in order to increase the amplitude of RTS (difference in current value between high and low current states). The probability of observing high and low current states were almost equal at V_{LG} = V_{RG} = -106mV, determining the bias condition of the time domain measurements. RTSs were observed with similar average lifetime as the one shown in Figure 3 but with higher average current (about 9nA), which were used to generate random numbers (part of data shown in Figure 7 (a)). The sampling interval of each measurement (t_{int}) was set to be 24ms, which is shorter than the average RTS lifetimes. 100001 points were taken, resulting in the total measurement time of 2400s in a single time domain measurement. After one time domain measurement was completed, 20s of hold time, sufficiently longer than the average lifetime of RTSs, was provided such that the auto-correlation between two adjacent measurements were safely removed.

In order to generate a sequence of random numbers from RTS, high current state ($|0\rangle$) was attributed to a digital number 1, while the low current state ($|1\rangle$ and $|2\rangle$) was designated to be a digital number 0. This can be considered as a classification problem of time-series data[29], and to achieve this classification, a certain mathematical model (classifier) needed to be established. Several methods can be proposed. The most simple method is to create a model every time after a measurement was completed (Figure 5 (a)). That is, after a single time domain measurement was completed, a histogram of I_d was created, and the analog-to-digital conversion (ADC) was performed based on the threshold value, determined from the histogram, that segregates one current state to the other. The merit of this procedure is that the ADC from RTS to random numbers can be achieved with high accuracy, as the best classifier is produced for a given time domain measurement. However, the problem of this method is that real-time random number generation is not possible, as the random number is returned as a result of the post processing.

In order to overcome this issue, initialisation of a classifier is necessary. That is, by using several time domain measurements at the same bias condition as 'training data', a classifier (for example, allocate 1 if I_d exceeds a certain value determined from training data, else 0, and the sampling time twice longer than average lifetime of RTS in training data) was created, before the model was applied to a new RTS ('test data') to generate a sequence of random numbers (Figure 5 (b)). Dividing dataset into training and test data, building a mathematical model before applying it to the test data is a typical method used in the context to supervised learning. The clear advantage of this method over the aforementioned one is that the real time random number generation is possible. That is, after the classifier is established, a sequence of random numbers can be generated by applying the classifier to the measurement as the instrument records I_d over time. A disadvantage of this method is that as the classifier is optimised to the training data and not the test data, there may be a misclassification of current states. This misclassification is unavoidable in classification problems. However, the accuracy can be im-

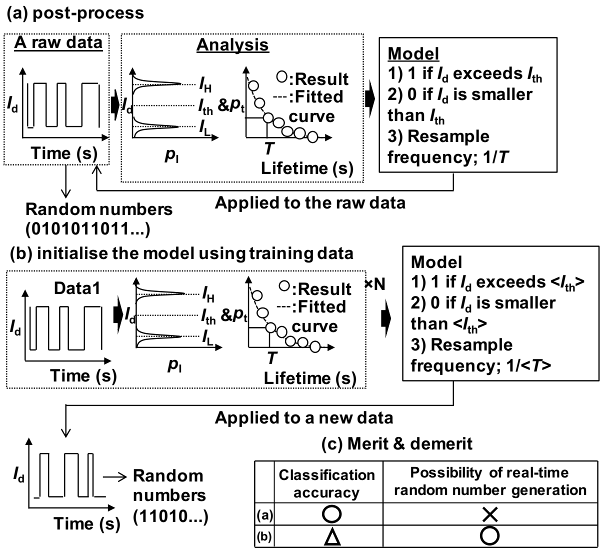

(a) post-process
A raw data
I_d
Time (s)
Analysis
I_H
I_{th} &p_t
I_L
p_I
O:Result
---:Fitted curve
T
Lifetime (s)
Model
1) 1 if I_d exceeds I_{th}
2) 0 if I_d is smaller than I_{th}
3) Resample frequency; 1/T
Random numbers (0101011011...)
Applied to the raw data
(b) initialise the model using training data
Data1
×N
Model
1) 1 if I_d exceeds <I_{th}>
2) 0 if I_d is smaller than <I_{th}>
3) Resample frequency; 1/<T>
Applied to a new data
Random numbers (11010...)
(c) Merit & demerit
Classification accuracy
Possibility of real-time random number generation
(a) ○ ×
(b) △ ○

FIG. 5. Schematics for describing procedures to generate random number from random telegraph signals. (a) shows the protocol when random numbers are generated from a single measurement of random telegraph signal (post-processing). (b) shows the protocol when the classification model is generated from several measurements at the same bias condition before it is applied to a new result to generate random numbers (initialise the model using training data). (c) summarises the advantage and disadvantage of the two procedures described above.

proved by refining the model (taking into account a drift of average current due to bias temperature instability[35–38], for instance). The merit and demerit of both methods were summarised as a table in Figure 5 (c).

In the following, the procedure to generate a classifier is explained based on our data, whose flowchart is schematically shown in Figure 6. While the method itself can be used for real-time random number generation, in this paper real-time random number generation was emulated by using test data, rather than obtain more time domain measurements. Firstly, seven time domain measurements out of nine results (Data1,2...9) were used as training data (called trainingData1,2...7), and the other two measurements were reserved for emulating random number generation (called testData1,2). The selection of training and test data was not necessarily in the order of time, as all measurements were assumed to be independent due to the hold time of 20s. This means that there were ${}_9C_2$ = 36 combinations of training and test data, that can be used to evaluate randomness of the generated random numbers in section VI. For simplicity, one particular combination was focused to explain the procedure in detail. Out of trainingData1,2...7, one result was selected (trainingData1, for example) and four parameters that characterise the time domain measurements were extracted, which are $I_{\rm th}$, $I_{\rm max}$, $I_{\rm min}$ and T. The first three parameters were used for ADC, while T determines the re-sampling frequency of test data. The definition of these parameters are explained in detail in the next two paragraphs. Extraction of these parameters was achieved by creating two histograms, which are histogram of current values (calculated by a function called 'histcount1') and histogram of individual lifetimes (calculated by a function called 'histcount2').

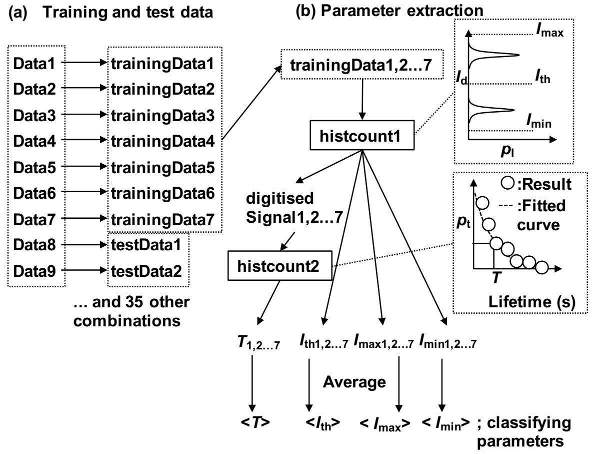


FIG. 6. Schematics to show how the training of the model was performed. (a) shows the division of the nine results (Data1,2...9) into training data (trainingData1,2...7) and test data (testData1 and 2). (b) shows the flowchart of parameter extraction from trainingData1,2...7. A function 'histcount1' generates a histogram of $I_{\rm d}$ value over time, outputting $I_{\rm th}$, $I_{\rm max}$, $I_{\rm min}$ and digitised signal ('digitisedSignal'). The digitised signal was further analysed by a function 'histcount2', which generates a histogram of individual lifetimes, which resulted in the extraction of T. This process was iterated for all trainingData1,2...7 and average was taken, resulting in the final outputs $\langle I_{\rm th}\rangle$, $\langle I_{\rm max}\rangle$, $\langle I_{\rm min}\rangle$ and $\langle T\rangle$

'histcount1' was defined to accept one time domain measurement (trainingData1, for example) as an input and firstly calculate a probability distribution to observe a certain current value of $I_{\rm d}$ over the measurement time. Due to the presence of RTS, two major peaks were observed in the histogram (Fig 3 (b). These two peaks were then fitted by a double Gaussian function

$$f(I_{\rm d}) = a_1 \exp[-\{(I_{\rm d}-b_1)/c_1\}^2] + a_2 \exp[-\{(I_{\rm d}-b_2)/c_2\}^2], \qquad (1)$$

where a_1, b_1, c_1, a_2, b_2 and c_2 are all fitting parameters. In particular, b_1 and b_2 are the medians of the two Gaussian functions fitted to the probability distribution of current values, representing the average value for high and low current state. b_1, c_1, b_2 and c_2 have the dimension of Ampere, while a_1 and a_2 are dimensionless. Then, b_1 and b_2 were used to determine the threshold current value to classify the current state (called $I_{\rm th}$), and maximum and minimum current values accepted in the given time domain measurement (called $I_{\rm max}$ and $I_{\rm min}$, respectively) to remove outliers. $I_{\rm th}$ was simply defined as an average of b_1 and b_2

$$I_{\rm th} = (b_1 + b_2)/2, \qquad (2)$$

while $I_{\rm max}$ and $I_{\rm min}$ were defined as

$$I_{\rm max} = I_{\rm th} + 2|b_1 - b_2| \qquad (3)$$

$$I_{\rm min} = I_{\rm th} - 2|b_1 - b_2|, \qquad (4)$$

respectively. This means that when I_d exceeded I_max, the digital number would not be allocated and NaN was returned instead of 1. Similarly, when I_d was lower than I_min, NaN was given instead of 0. When I_d is bigger than I_th and smaller than I_max, a digital number of 1 was designated, while when I_d is smaller than I_th and larger than I_min, a digital number of 0 would be given. This digitisation was performed to obtain a sequence of digital numbers, called ‘digitisedSignal’, which constituted, together with I_th, I_max and I_min, the output of function ‘histcount1’. Figure 7 (a) shows part of Data8, I_th, I_max, I_min calculated from ‘histcount1’ with Data8 as an input, and ‘digitisedSignal’ plotted over the raw data. When an outlier was observed, like the points around t=1910s, NaN was given instead of 0. This outlier might have been possibly caused by another RTS with lower probability to be observed compared to the one that has been discussed so far[15].

‘histcount2’ was designed to accept ‘digitisedSignal’ as an input and firstly calculate a probability distribution of individual lifetime of one current state (high current state, for example) of a given training data (Data8, for example)[15]. As the capture and emission process of a single electron in a QD is considered to follow Poisson process, the distribution was expected to be fitted by an exponential function[15]. Figure 7 (b) shows a histogram of individual lifetime of high current state in Data8 as a training data, and indeed the probability to observe the same current state became exponentially smaller. The histogram was fitted by a simple exponential function

$$P_\mathrm{High}(t) = \frac{1}{\langle\tau_\mathrm{High}\rangle}\exp\left\{\frac{-t}{\langle\tau_\mathrm{High}\rangle}\right\}, \tag{5}$$

where $\langle\tau_\mathrm{High}\rangle$ is a fitting parameter, whose physical meaning can be understand as follows; after $\langle\tau_\mathrm{High}\rangle$, the probability for high current state not making a transit to low current state is $1/e \sim 37\%$[15]. In this sense, $\langle\tau_\mathrm{High}\rangle$ represents the average lifetime of high current state in a given time domain measurement. Also, this value can be understood as a measure to evaluate the autocorrelation of the RTS signal. That is, if RTS was monitored longer than $\langle\tau_\mathrm{High}\rangle$, the probability for I_d not making a transit from high current state to low current state can be considered as sufficiently small. The same histogram was calculated for low current state, and $\langle\tau_\mathrm{Low}\rangle$, average lifetime of low current state, were obtained;

$$P_\mathrm{Low}(t) = \frac{1}{\langle\tau_\mathrm{Low}\rangle}\exp\left\{\frac{-t}{\langle\tau_\mathrm{Low}\rangle}\right\}. \tag{6}$$

$\langle\tau_\mathrm{High}\rangle$ and $\langle\tau_\mathrm{Low}\rangle$ were almost identical as the probability to observe high and low current states were nearly equal (see Figure 7 (a) and 10 (c)). However, it is virtually impossible to balance the probabilities to be exactly 50% with infinite precision, due to the limited resolution of voltage source (1μV). This means that either $\langle\tau_\mathrm{High}\rangle$ or $\langle\tau_\mathrm{Low}\rangle$ would be slightly longer than the other. For consistency, the longer one was defined as a typical lifetime of RTS;

$$\langle\tau\rangle = \max\{\langle\tau_\mathrm{High}\rangle, \langle\tau_\mathrm{Low}\rangle\} \tag{7}$$

The time-scale parameter extracted from histcount2 would be used for re-sampling of test data for RNG, explained in the next section V. For this purpose, the re-sampling interval should be sufficiently longer than a typical lifetime of RTS in order to reduce autocorrelation. Therefore, $\langle\tau\rangle$ was multiplied twice such that the probability for one current state did not make a transit to the other current state was less than $1/e^2 \sim 14\%$;

$$T = 2\langle\tau\rangle, \tag{8}$$

which is an output of ‘histcount2’.

After the four parameters were extracted from trainingData1 by ‘histcount1’ and ‘histcount2’, the same procedure was iterated for the remaining six time domain measurements (trainingData2,3...7). This resulted in seven sets of four parameters, and finally the average of each parameter was calculated, resulting in $\langle I_\mathrm{th}\rangle$, $\langle I_\mathrm{max}\rangle$, $\langle I_\mathrm{min}\rangle$ and $\langle T\rangle$;

$$\langle I_\mathrm{th}\rangle = \frac{1}{k}\sum_{i=1}^{k} I_{\mathrm{th},i}, \tag{9}$$

$$\langle I_\mathrm{max}\rangle = \frac{1}{k}\sum_{i=1}^{k} I_{\mathrm{max},i}, \tag{10}$$

$$\langle I_\mathrm{min}\rangle = \frac{1}{k}\sum_{i=1}^{k} I_{\mathrm{min},i}, \tag{11}$$

$$\langle T\rangle = \frac{1}{k}\sum_{i=1}^{k} T_i, \tag{12}$$

where k is the number of training data (7 in this case), $I_{\mathrm{th},i}$, $I_{\mathrm{max},i}$, $I_{\mathrm{min},i}$ and T_i are I_th, I_max, I_min and T of trainingData i, respectively. Note that ‘$\langle T\rangle$/2’ is average of typical lifetimes of RTSs in trainingData1,2...7. These four parameters underpin the classifier for extracting random numbers from RTS.

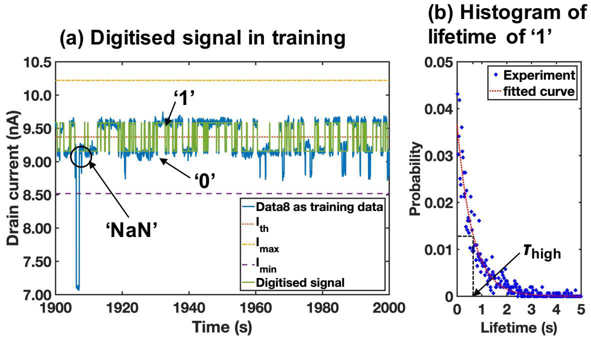


FIG. 7. Examples of the output of functions ’histcount1’ and ’histcount2’. (a) shows an example of output of histcount1 when Data8 was used as an input (training data). In (a), part of measurement result (Data8), I_th, I_max, I_min determined from Data8, and the digitised signal generated from Data 8 by the classification parameters (I_th, I_max, I_min) were shown. When outliers were observed, histcount1 returned NaN, as shown in the inset. (b) shows a histogram of individual lifetimes of high current state in Data8, generated by histcount2. Blue filled circles are the experimental results, while the red dotted line is the exponential function fitted to the results, allowing one to extract the average lifetime of high current states τ_high.

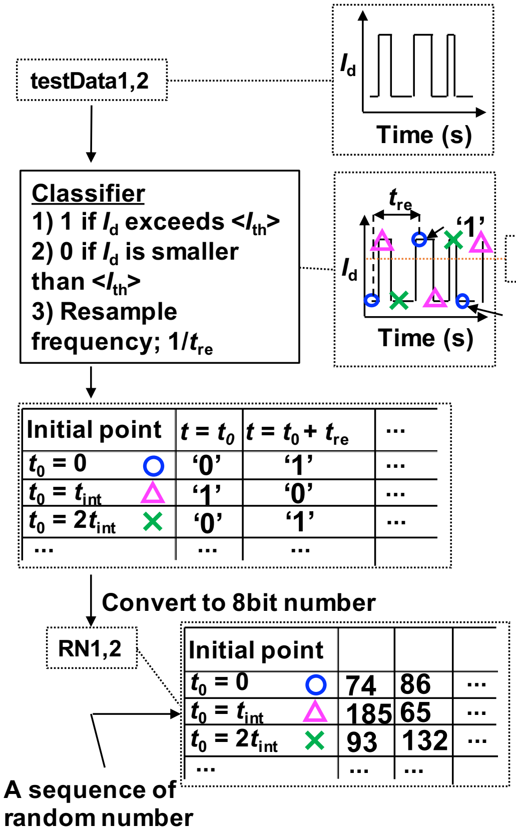
V. RANDOM NUMBER GENERATED FROM RANDOM TELEGRAPH SIGNALS OBSERVED IN THE SINGLE ELECTRON PUMP DEVICE

Using the classifier determined from the training data via histcounts1 and histcounts2, finally a sequence of random numbers was extracted from test data. A schematic of this procedure is shown in Figure 8. The extraction procedure is 1) allocate a digital number ‘1’ if $\langle I_{\mathrm{max}}\rangle > I_{\mathrm{d}} > \langle I_{\mathrm{th}}\rangle$ was satisfied 2) allocate ‘0’ if $\langle I_{\mathrm{th}}\rangle > I_{\mathrm{d}} > \langle I_{\mathrm{min}}\rangle$ was satisfied 3) allocate NaN if I_{d} exceeded $\langle I_{\mathrm{max}}\rangle$ or I_{d} was smaller than $\langle I_{\mathrm{min}}\rangle$. 4) resample ‘digitisedSignal’ with an interval of

$$t_{\mathrm{re}} = \lfloor \langle T\rangle / t_{\mathrm{int}} \rfloor t_{\mathrm{int}}, \tag{13}$$

where t_{int} is a measurement interval (24ms), and $\lfloor\rfloor$ is a floor function, fixing the value of a real number $\langle T\rangle/t_{\mathrm{int}}$ to the closest integer equal to or less than $\langle T\rangle/t_{\mathrm{int}}$. The re-sampling can only be achieved with an interval of $\langle T\rangle$ when $\langle T\rangle$ happened to be an integer multiple of t_{int}. Since $\langle T\rangle$ is not necessarily an integer multiple of t_{int}, the floor function was practically necessary. The purpose of the re-sampling step was to remove autocorrelation between one digital number extracted from one point to the other[14]. When RTS is sampled with interval sufficiently shorter than the average lifetime of RTS, it is likely to observe the same current state at the next measurement point, which is obvious from the definition of lifetime of one current state. In order to extract a sequence of random numbers from RTS, it is required to be sampled with interval sufficiently longer than the average lifetime such that a digital number at one point is unpredictable from the previous digital number. Each T calculated from training data was defined to be twice of the typical lifetime of RTS, and therefore, on average, $\lfloor\langle T\rangle/t_{\mathrm{int}}\rfloor t_{\mathrm{int}}$ can be considered to be sufficiently longer than the lifetime of testData.

Note that in this paper the real-time random number generation is emulated and not actually performed, and the test data has the same number of measurement points as the training data. Depending on the initial point of re-sampling was taken, there were $\lfloor\langle T\rangle/t_{\mathrm{int}}\rfloor$ ways of extracting a sequence of binary numbers from a test data (a circle, triangle and cross mark in Figure 8). This is because the selection of the initial point was purely arbitrary; as the test data is considered to be independent of the training data, the measurement could have started at time $t_0 = t_{\mathrm{int}}$, $t_0 = 2t_{\mathrm{int}}$, ... $t_0 = (\lfloor\langle T\rangle/t_{\mathrm{int}}\rfloor - 1)t_{\mathrm{int}}$. This means that sequences of random numbers starting at $t_0 = t_{\mathrm{int}}$, $t_0 = 2t_{\mathrm{int}}$, ... $t_0 = (\lfloor\langle T\rangle/t_{\mathrm{int}}\rfloor - 1)t_{\mathrm{int}}$ were all equally likely and should not be discarded. Therefore, all sequences were saved for further analysis. The number of sequences depended on which data (Data1,2...9) was used for test data, particularly the number of NaN in the digitised signal of the test data. In our case, $\lfloor\langle T\rangle/t_{\mathrm{int}}\rfloor \sim 55$ sequences could be extracted from one time domain measurement. Finally, eight adjacent binary numbers were combined to form one decimal number (from 0 to 255), resulting in $\lfloor\langle T\rangle/t_{\mathrm{int}}\rfloor$ sequences of 8-bit decimal random numbers, which was saved as a matrix called RN1 (one row of RN1 corresponds to a sequence of random numbers). To implement the real-time random number generation, time domain measurement should be taken with an interval of $\lfloor\langle T\rangle/t_{\mathrm{int}}\rfloor t_{\mathrm{int}}$ in the first place, before the classifier was applied for ADC. This procedure was repeated for testData2, and a matrix of random numbers with the same size as the one generated from testData1 was obtained, called RN2.


testData1,2
I_d
Time (s)
Classifier
1) 1 if I_d exceeds <I_{th}>
2) 0 if I_d is smaller than <I_{th}>
3) Resample frequency; 1/t_{re}
t_{re}
‘1’
I_d
<I_{th}>
‘0’
Time (s)
Convert to 8bit number
RN1,2
A sequence of random number

Initial point	$t = t_0$	$t = t_0 + t_{re}$	...
$t_0 = 0$ ○	‘0’	‘1’	...
$t_0 = t_{int}$ △	‘1’	‘0’	...
$t_0 = 2t_{int}$ ✕	‘0’	‘1’	...
...	...	...	

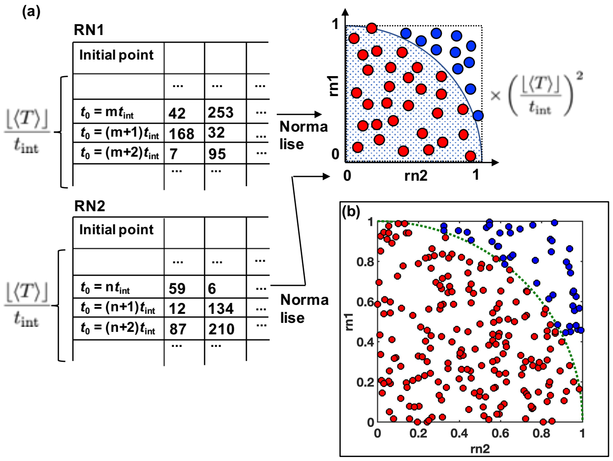
Initial point			
$t_0 = 0$ ○	74	86	...
$t_0 = t_{int}$ △	185	65	...
$t_0 = 2t_{int}$ ✕	93	132	...
...	...	...	

FIG. 8. A flow chart of extracting sequences of random number from testData1 and 2. The accurate description of the classifier is given in section V. testData1 and 2 were digitised by the classifier and also reorganised in a matrix depending on the initial sampling points. Binary random numbers were finally converted to 8bit decimal numbers, and two matrices called RN1 and RN2 were produced from testData1 and 2.

VI. EVALUATION OF RANDOMNESS GENERATED FROM RANDOM TELEGRAPH SIGNALS

In the section IV and V, the algorithm to generate random numbers from RTSs was introduced, and two matrices containing the sequences of random numbers (RN1 and RN2) were generated as an output. In this section, the randomness of the random numbers is evaluated by estimating the value of π using Monte Carlo method. This estimation of π is generally achieved as follows. Firstly, two sequences of N random numbers distributed in [0,1] were prepared using any random number generator. These two sequences of random number can be used as a coordinate to designate a point in the x-y plane ($0 \leq x \leq 1, 0 \leq y \leq 1$). Then, a number of points that satisfy the condition $x^2 + y^2 < 1$ were counted, which is an estimate of $\pi/4$. Because a random number generator should generate a different sequence of random numbers every time it runs, this estimate of π will also be different depending on the random number used as inputs. The important point is the distribution of a large number of estimated values of π, and the quality of random number can be considered as good when the distribution has a median around the true value (3.14...) with smaller deviation from the median. The accuracy of the median is determined by the randomness of the random number, while the precision of the estimation, gauged by the deviation from the median, is limited by the number of elements in a sequence (number of points in the x-y plane). This means that if a longer sequence was used as an input, it is more likely that the distribution of estimated values would be smaller.

Random numbers generated from RTS was then used as inputs of this procedure to calculate π (Figure 9 (a)). One sequence of random numbers was selected from both RN1 and RN2, and these random numbers were normalised such that all values would be fit in [0,1], called rn1 and rn2. rn1 and rn2 were then used as a coordinate of a point in the x-y plane (rn1 for x, rn2 for y, for example), creating a scatter plot in the area $0 \leq x \leq 1, 0 \leq y \leq 1$. Finally π was estimated by counting number of points in $x^2 + y^2 < 1$. This estimation was repeated for all the combinations that can be taken from RN1 and RN2, which means that π can be calculated $\lfloor \langle T \rangle / t_{\mathrm{int}} \rfloor^2$ times from RN1 and RN2. Figure 9 (b) shows one of the examples of a scatter plot generated from random numbers originating from Data3 and Data8. The estimated value of π from this particular distribution resulted in 3.15.


(a)
RN1
$\frac{\lfloor\langle T\rangle\rfloor}{t_{\mathrm{int}}}$
Normalise
RN2
$\frac{\lfloor\langle T\rangle\rfloor}{t_{\mathrm{int}}}$
Normalise
1
rn1
0
0
rn2
1
$\times \left(\frac{\lfloor\langle T\rangle\rfloor}{t_{\mathrm{int}}}\right)^2$
(b)
1
0.8
0.6
0.4
0.2
0
rn1
0
0.2
0.4
0.6
0.8
1
rn2

RN1

Initial point			
	...	...	...
$t_0 = \mathrm{m}t_{\mathrm{int}}$	42	253	...
$t_0 = (\mathrm{m}+1)t_{\mathrm{int}}$	168	32	...
$t_0 = (\mathrm{m}+2)t_{\mathrm{int}}$	7	95	...
	...	...	

RN2

Initial point			
	...	...	...
$t_0 = \mathrm{n}t_{\mathrm{int}}$	59	6	...
$t_0 = (\mathrm{n}+1)t_{\mathrm{int}}$	12	134	...
$t_0 = (\mathrm{n}+2)t_{\mathrm{int}}$	87	210	...
	...	...	

FIG. 9. (a) shows the procedure to estimate π from random numbers in RN1 and RN2. One sequence of random numbers were chosen from both RN1 and RN2, normalised in [0,1], and used as a coordinate to designate a point in the x-y plane. Red filled circles are the points in the area $x^2 + y^2 < 1$, while blue filled circles were points outside. (b) shows one of the examples of rn1 and rn2 taken from RN1 and RN2, generated from Data3 and Data8 as test data.

Figure 10 (a) shows two examples of the distribution of estimated π, which are both compared with the values of π estimated from Monte Carlo method using uniform random number generated by Mersenne twister (MT). The blue points in Figure 10 (a) are probability distribution of calculated values of π using Data1,2,4,5,6,7,9 to generate the classifier (trainingData) and Data3,8 to generate random numbers. When fitted by a Gaussian function, the median was 3.16 and the variance was 0.0750, which was comparable with the result obtained from uniformly generated random number by MT as input (plotted as orange dashed line in Figure 10 (a)). On the other hand, magenta points in Figure 10 (a) are the one using Data 3,4,5,6,7,8,9 for the classifier and Data1,2 for random numbers. As can be seen, the median of the distribution was off from the true value. This implies that the numbers generated from Data1,2 by the classifier based on Data3,4...9 were not uniform, which means that a number of smaller values (0,1,2...) was larger than that of bigger values (...253, 254, 255).

In order to seek the source of non-uniformity in random numbers generated from Data1,2 as test data, the validity of classification parameters were investigated. Figure 10 (b), (c), (d) and (e) show the histograms of Data3, Data8, Data1 and Data2 (light-blue lines) with corresponding I_{th} and $\langle I_{\mathrm{th}} \rangle$ values. These values are highlighted by blue solid line and magenta dotted line, respectively. I_{th} is defined to be an average of two medians of double Gaussian functions fitted to a given histogram, meaning that the digitised signal generated using I_{th} can be used as a reference. On the other hand, as $\langle I_{\mathrm{th}} \rangle$ is an average of I_{th} of multiple training data, which is not optimised to a test data, chances are that a current state can be misclassified to a different one due to fluctuation of measured current values around the median. Therefore, the difference between I_{th} and $\langle I_{\mathrm{th}} \rangle$ can be used as a metric to characterise the extent of misclassification;

$$d_{ij} = \langle I_{\mathrm{th}} \rangle_{ij} - I_{\mathrm{th},i}, \tag{14}$$

where i and j are natural numbers that satisfy $1 \leq i \neq j \leq 9$, $I_{\mathrm{th},i}$ is I_{th} for Data'i' and $\langle I_{\mathrm{th}} \rangle_{ij}$ is $\langle I_{\mathrm{th}} \rangle$ when Data'i' and Data'j' were used as test data. Positive d_{ij} means that misclassification of '1' as '0' frequently occurred when Data'i' was used as test data alongside with Data'j'. Likewise, negative d_{ij} implies that '0' were often misclassified as '1' when Data'i' was used as test data alongside with Data'j'. As shown in Figure 10 (b) and (c), d_{38} and d_{83} has a different sign, while d_{12} and d_{21} were both positive, seen in Figure 10 (d) and (e). This means that because more misclassification towards '0' occurred in both Data1 and Data2, RN1 and RN2 has a larger number of smaller values, leading to a larger number of points in the x-y plane that satisfy $x^2 + y^2 < 1$ and hence larger estimated values of π. In the case of Data3 and Data8, certainly there were misclassification in both cases, though the opposite sign of d_{38} and d_{83} helped to mask the non-uniformity in generated random number in the estimated values of π. This result suggests that even if the uniformity of a sequence of random numbers is not perfect, by knowing the deviation with its sign (d_{ij}, for example) and cancelling the deviations by combining two or three sequences of random numbers ($d_{\Sigma,ij}$), total uniformity of the points in higher dimension (2D,3D...) may be improved.

Combined with the distribution shown in Figure 10 (a), the randomness of points in the x-y plane can be evaluated by the

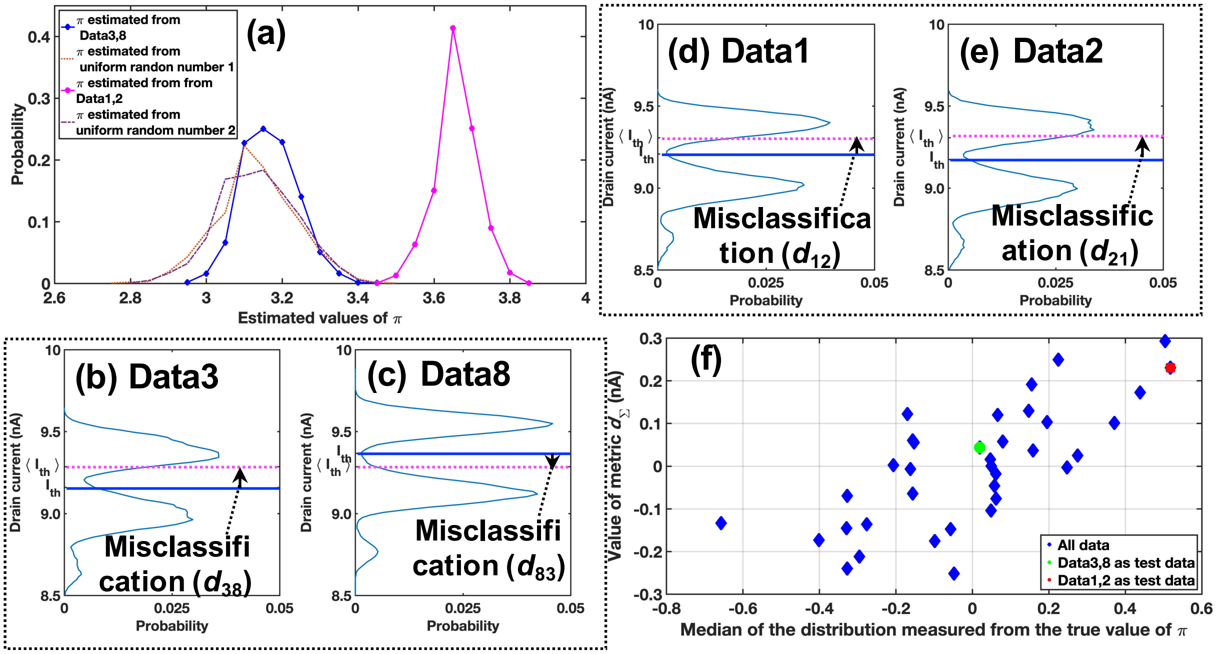


FIG. 10. Analysis on the estimated values of π. (a) shows the distributions of estimated values of π using two different test data. Blue solid diamonds are the one obtained from Data3 and Data8 as test data, while magenta filled circles are the one obtained from Data1 and Data2 as test data. Orange dotted line and purple dotted-dashed line are the ones calculated using uniform random number generated by Mersenne twister. (b) and (c) show the histograms of I_d of Data3 and Data8, respectively, (light-blue line). Blue solid lines in (b) and (c) denote threshold current (I_th) in each data, while magenta dotted lines in (b) and (c) show the average threshold current in training data ($\langle I_\mathrm{th}\rangle$). (d) and (e) show the histogram of I_d of Data1 and Data2, respectively. (f) Values of metric d_Σ against the median of the distributions measured from the true value of π.

sum of d_{ij} and d_{ji};

$$d_{\Sigma,ij} = d_{ij} + d_{ji}; \tag{15}$$

Figure 10 (f) shows the values of this metric $d_{\Sigma,ij}$ against the median of the distribution of estimated values of π, measured from its true value. Blue diamonds denote the results from all the combinations of choosing two test data from nine measurement results, and a green filled circle highlights the case when Data3,8 were used as test data, while the red filled circle does the case when Data1,2 were used as test data. A clear pattern can be seen in this scatter plot, where the median of the distribution deviate from the true value when the value of $|d_{\Sigma,ij}|$ becomes larger. This suggests that the metric $d_{\Sigma,ij}$ can be used as a predictor of how good the estimation would be. In other words, if the value of $d_{\Sigma,ij}$ is small, the median of the distribution is expected to be closer to the true value than otherwise.

Finally, in order to evaluate the overall quality of the random number generated from our device, a histogram of the estimated values of π from all possible combinations was calculated, which is shown in Figure 11. In total, π was calculated 80855 times using the Monte Carlo method. The blue filled circles are the data points, while the blue dotted line is a Gaussian fit with the median of 3.14 and the variance of 0.273. This distribution can be broken down into individual 36 histograms, two of which are shown in Fig 10 (a). Therefore, the variance of the total distribution is determined by distributions like the one obtained from Data1,2, which miscalculate values of π due to misclassification. By reducing the misclassification (improving the learning algorithms, for example[29]), the variance can also be smaller. In the same figure, the result from the uniform random number generated by MT is also shown, which has the median at 3.13 and the variance of 0.105, the latter of which is smaller than the one obtained from RTS, as expected.

VII. CONCLUSION

In this paper, random number generation based on random telegraph signals observed in a Si SEP device was discussed. RTSs was observed in one of the SEP devices, which showed a tunability of current states as expected[25,26]. The observed RTS has three current levels, whose physical origin was attributed to smaller energy gap between the ground state and the first excited state. In order to achieve real-time random number generation, the classifier to perform analog-to-digital conversion should be learned from previous measurement results, rather than a result from which random numbers are to be extracted. Therefore, such an algorithm was developed and introduced in Sec.IV and V in detail. After the classifier was established using training data, random numbers were extracted from a new test data, whose uniformity was benchmarked by calculating values of π using Monte Carlo method

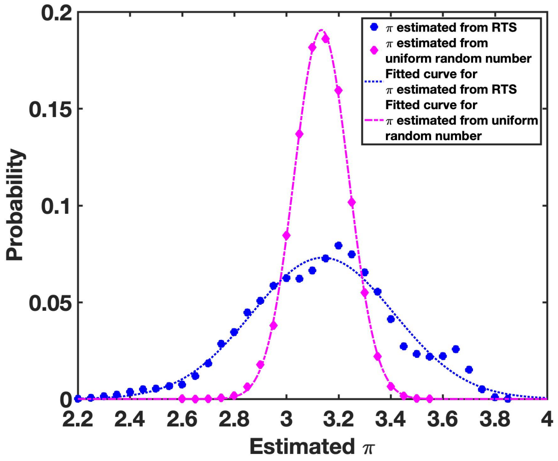


FIG. 11. Comparison between random number generated from RTS and from Mersenne twister (a PRNG). The blue solid circles are the distribution of calculated π from all data, while the blue dotted line show Gaussian fit to the distribution. The magenta filled circles are the distribution obtained from MT, while the magenta dash-dotted line show Gaussian fit to the distribution.

and compared with the result obtained from pseudo random numbers generated by MT. It turned out that the difference between the classifying parameters obtained from previous results and generated from the new result can serve as a predictor to evaluate the extent of misclassification and hence the randomness in the generated random numbers.

RTSs have been considered as a reliability issue in modern CMOS devices[17–23] mainly because the variability of RTSs cannot be controlled across wafers or fabrication lots, and also parameters that characterise RTSs seem to be not correlated with each other[15,28,39]. This certainly hinders the application of RTSs as HRNG, since the observation of RTSs is rare in the first place, and additionally the clock frequency at which the sampling would be made cannot be determined *a priori*. The device and algorithm shown in this paper addressed both difficulties, in a sense that SEP devices reliably exhibited RTSs[25,26] and the unpredictability of average lifetimes was harnessed by implementing 'supervised learning'.

While the bit rate of this proposed RNG turned out to be relatively slow, this was limited by the characteristics of RTSs used in this study. Traditionally, RTSs with average lifetime of less than μs were rarely reported in Si devices[40–45]. However, several studies show that the trapping and de-trapping of an electron in a QD or in a trap at Si-Silicon dioxide (SiO_2) interface can be cycled at a frequency of GHz[46,47]. This means that, combined with the CMOS compatibility, utilising RTSs as a source of randomness can be a promising candidate for future hardware security.

CONFLICT OF INTEREST STATEMENT

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

AUTHOR CONTRIBUTIONS

M.K.H, Z.L, and S.S fabricated and prepared samples for measurement. F.L, K.I, Y.T, H.R and S.S established measurement setup. M.K.H, F.L, K.I, J.W.H and S.S performed measurements. K.I, I.T and S.S proposed the physical model of the observed random telegraph signals. K.I analysed the data, including the extraction of random numbers from random telegraph signals and evaluation of π using Monte Carlo method. K.I, M.S, F.L and S.S discussed the randomness of the random numbers. K.I prepared figures and drafted manuscript. All authors participated in discussion.

DATA AVAILABILITY STATEMENT

The data that supports the findings of this study are openly available in ePrints Soton, the University of Southampton Institutional Research Repository (https://doi.org/10.5258/SOTON/D1488)[48]

ACKNOWLEDGMENTS

This work is supported by EPSRC Manufacturing Fellowship (EP/M008975/1), Lloyds Register Foundation International Consortium of Nanotechnology, and the Joint Research Project (e-SI-Amp (15SIB08)). This work is also supported by the European Metrology Programme for Innovation and Research (EMPIR) co-financed by the Participating States and from the European Union's Horizon 2020 research and innovation programme.

[1] L. Gong, M. A. Lomas, R. M. Needham, and J. H. Saltzer, "Protecting poorly chosen secrets from guessing attacks," IEEE Journal on Selected Areas in Communications **11**, 648–656 (1993).

[2] S. M. Bellovin and M. Merritt, "Encrypted key exchange: password-based protocols secure against dictionary attacks," in *Proceedings 1992 IEEE Computer Society Symposium on Research in Security and Privacy* (1992) pp. 72–84.

[3] Z. Zhao, Z. Dong, and Y. Wang, "Security analysis of a password-based authentication protocol proposed to ieee 1363," Theor. Comput. Sci. **352**, 280–287 (2006).

[4] D. R. Stinson, "Universal hashing and authentication codes," Designs, Codes and Cryptography **4**, 369–380 (1994).

[5] B. Halak, *Physically Unclonable Functions: From Basic Design Principles to Advanced Hardware Security Applications* (Springer, 2018).

[6] R. Rivest and S. Dusse, "The md5 message-digest algorithm," (1992).

[7] Huang Zhun and Chen Hongyi, "A truly random number generator based on thermal noise," in *ASICON 2001. 2001 4th International Conference on ASIC Proceedings (Cat. No.01TH8549)* (2001) pp. 862–864.

[8] J. F. Dynes, Z. L. Yuan, A. W. Sharpe, and A. J. Shields, "A high speed, postprocessing free, quantum random number generator," Appl. Phys. Lett. **93**, 031109 (2008).

[9] Z. Chai, W. Shao, W. Zhang, J. Brown, R. Degraeve, F. D. Salim, S. Clima, F. Hatem, J. F. Zhang, P. Freitas, *et al.*, "Gese-based ovonic threshold switching volatile true random number generator," IEEE Electron Device Letters **41**, 228–231 (2019).

[10] T. Harayama, S. Sunada, K. Yoshimura, P. Davis, K. Tsuzuki, and A. Uchida, "Fast nondeterministic random-bit generation using on-chip chaos lasers," Phys. Rev. A **83**, 031803 (2011).

[11] X. Chen, L. Wang, B. Li, Y. Wang, X. Li, Y. Liu, and H. Yang, "Modeling random telegraph noise as a randomness source and its application in true random number generation," EEE Trans. Comput.-Aided Des. Integr. Circuits Syst. **35**, 1435–1448 (2015).

[12] K. Uchida, T. Tanamoto, R. Ohba, S. Yasuda, and S. Fujita, "Single-electron random-number generator (RNG) for highly secure ubiquitous computing applications," in *IEDM Tech. Dig.* (IEEE, 2002) pp. 177–180.

[13] R. Brederlow, R. Prakash, C. Paulus, and R. Thewes, "A low-power true random number generator using random telegraph noise of single oxide-traps," in *2006 IEEE International Solid State Circuits Conference-Digest of Technical Papers* (IEEE, 2006) pp. 1666–1675.

[14] J. Brown, R. Gao, Z. Ji, J. Chen, J. Wu, J. Zhang, B. Zhou, Q. Shi, J. Crowford, and W. Zhang, "A low-power and high-speed true random number generator using generated rtn," in *Symp. VLSI Technol. Dig. Tech. Papers* (IEEE, 2018) pp. 95–96.

[15] E. Simoen and C. L. Claeys, *Random telegraph signals in semiconductor devices* (IOP Publishing Limited, 2016).

[16] K. Ralls, W. Skocpol, L. Jackel, R. Howard, L. Fetter, R. Epworth, and D. Tennant, "Discrete resistance switching in submicrometer silicon inversion layers: Individual interface traps and low-frequency (1/f) noise," Phys. Rev. Lett. **52**, 228 (1984).

[17] N. Tega, H. Miki, M. Yamaoka, H. Kume, T. Mine, T. Ishida, Y. Mori, R. Yamada, and K. Torii, "Impact of threshold voltage fluctuation due to random telegraph noise on scaled-down SRAM," in *Proc. 46th Ann. Int. Reliab. Phys. Symp. - IRPS08* (IEEE, 2008) pp. 541–546.

[18] M. Yamaoka, H. Miki, A. Bansal, S. Wu, D. Frank, E. Leobandung, and K. Torii, "Evaluation methodology for random telegraph noise effects in SRAM arrays," in *IEDM Tech. Dig.* (IEEE, 2011) pp. 745–748.

[19] K. Takeuchi, T. Nagumo, K. Takeda, S. Asayama, S. Yokogawa, K. Imai, and Y. Hayashi, "Direct observation of rtn-induced sram failure by accelerated testing and its application to product reliability assessment," in *Symp. VLSI Technol. Dig. Tech. Papers* (IEEE, 2010) pp. 189–190.

[20] K. Fukuda, Y. Shimizu, K. Amemiya, M. Kamoshida, and C. Hu, "Random telegraph noise in flash memories-model and technology scaling," in *IEDM Tech. Dig.* (IEEE, 2007) pp. 169–172.

[21] N. Tega, H. Miki, T. Osabe, A. Kotabe, K. Otsuga, H. Kurata, S. Kamohara, K. Tokami, Y. Ikeda, and R. Yamada, "Anomalously large threshold voltage fluctuation by complex random telegraph signal in floating gate flash memory," in *IEDM Tech. Dig.* (IEEE, 2006) pp. 218–221.

[22] H. Kurata, K. Otsuga, A. Kotabe, S. Kajiyama, T. Osabe, Y. Sasago, S. Narumi, K. Tokami, S. Kamohara, and O. Tsuchiya, "Random telegraph signal in flash memory: Its impact on scaling of multilevel flash memory beyond the 90-nm node," IEEE J. Solid-State. Circuits. **42**, 1362–1369 (2007).

[23] S.-R. Li, Y.-L. R. Lu, W. McMahon, Y.-H. Lee, and N. Mielke, "RTS and 1/f noise in flash memory," in *Symp. VLSI Technol. Dig. Tech. Papers* (IEEE, 2007) pp. 56–57.

[24] Z. Li, M. Sotto, F. Liu, M. K. Husain, H. Yoshimoto, Y. Sasago, D. Hisamoto, I. Tomita, Y. Tsuchiya, and S. Saito, "Random telegraph noise from resonant tunnelling at low temperatures," Sci. Rep. **8**, 250 (2018).

[25] F. Liu, K. Ibukuro, M. K. Husain, Z. Li, J. Hillier, I. Tomita, Y. Tsuchiya, H. Rutt, and S. Saito, "Manipulation of random telegraph signals in a silicon nanowire transistor with a triple gate," Nanotechnology **29**, 475201 (2018).

[26] K. Ibukuro, M. K. Husain, Z. Li, J. Hillier, F. Liu, Y. Tsuchiya, H. N. Rutt, and S. Saito, "Single electron memory effect using random telegraph signals at room temperature," Front. Phys. **7**, 152 (2019).

[27] K. Ibukuro, J. W. Hillier, F. Liu, M. K. Husain, Z. Li, I. Tomita, Y. Tsuchiya, H. N. Rutt, and S. Saito, "Random telegraph signals caused by a single dopant in a metal–oxide–semiconductor field effect transistor at low temperature," AIP Advances **10**, 055025 (2020).

[28] A. Yonezawa, R. Kuroda, A. Teramoto, T. Obara, and S. Sugawa, "A statistical evaluation of effective time constants of random telegraph noise with various operation timings of in-pixel source follower transistors," in *Image Sensors and Imaging Systems 2014*, Vol. 9022 (International Society for Optics and Photonics, 2014) p. 90220F.

[29] N. J. Lambert, A. A. Esmail, M. Edwards, A. J. Ferguson, and H. G. L. Schwefel, "Random telegraph signal analysis with a recurrent neural network," Phys. Rev. E **102**, 012312 (2020).

[30] O. Tabata, R. Asahi, H. Funabashi, K. Shimaoka, and S. Sugiyama, "Anisotropic etching of silicon in tmah solutions," Sens. Actuators A **34**, 51–57 (1992).

[31] A. Merlos, M. Acero, M. Bao, J. Bausells, and J. Esteve, "Tmah/ipa anisotropic etching characteristics," Sens. Actuators A **37**, 737–743 (1993).

[32] K. K. Lee, D. R. Lim, L. C. Kimerling, J. Shin, and F. Cerrina, "Fabrication of ultralow-loss Si/SiO 2 waveguides by roughness reduction," Opt. Let. **26**, 1888–1890 (2001).

[33] K. Nishiguchi, Y. Ono, and A. Fujiwara, "Single-electron counting statistics of shot noise in nanowire si metal-oxide-semiconductor field-effect transistors," Appl. Phys. Lett. **98**, 193502 (2011).

[34] K. Nishiguchi, Y. Ono, and A. Fujiwara, "Single-electron thermal noise," Nanotechnology **25**, 275201 (2014).

[35] J. Zhang and W. Eccleston, "Positive bias temperature instability in MOSFETs," IEEE Trans. Elec. Dev. **45**, 116–124 (1998).

[36] D. K. Schroder and J. A. Babcock, "Negative bias temperature instability: Road to cross in deep submicron silicon semiconductor manufacturing," J. Appl. Phys. **94**, 1–18 (2003).

[37] T. Grasser, H. Reisinger, W. Goes, T. Aichinger, P. Hehenberger, P.-J. Wagner, M. Nelhiebel, J. Franco, and B. Kaczer, "Switching oxide traps as the missing link between negative bias temperature instability and random telegraph noise," in *IEDM Tech. Dig.* (IEEE, 2009) pp. 729–732.

[38] T. Grasser, "Stochastic charge trapping in oxides: From random telegraph noise to bias temperature instabilities," Microelectron. Reliab. **52**, 39–70 (2012).

[39] T. Nagumo, K. Takeuchi, T. Hase, and Y. Hayashi, "Statistical characterization of trap position, energy, amplitude and time constants by RTN measurement of multiple individual traps," in *IEDM Tech. Dig.* (IEEE, 2010) pp. 28–3.

[40] A. Whitcombe, S. Taylor, M. Denham, V. Milovanović, and B. Nikolić, "On-chip i–v variability and random telegraph noise characterization in 28 nm cmos," in *2016 46th European Solid-State Device Research Conference (ESSDERC)* (2016) pp. 248–251.

[41] A. S. Rouf, Z. Celik-Butler, F.-C. Hou, S. Tang, and G. Mathur, "Two types of E' centers as gate oxide defects responsible for hole trapping and random telegraph signals in pmosfets," IEEE Transactions on Electron Devices **65**, 4527–4534 (2018).

[42] K. Abe, S. Sugawa, S. Watabe, N. Miyamoto, A. Teramoto, Y. Kamata, K. Shibusawa, M. Toita, and T. Ohmi, "Random telegraph signal statistical analysis using a very large-scale array teg with 1m mosfets," in *2007 IEEE Symposium on VLSI Technology* (IEEE, 2007) pp. 210–211.

[43] A. Yonezawa, A. Teramoto, T. Obara, R. Kuroda, S. Sugawa, and T. Ohmi, "The study of time constant analysis in random telegraph noise at the subthreshold voltage region," in *Proc. 51th Ann. Int. Reliab. Phys. Symp. - IRPS13* (IEEE, 2013) pp. XT–11.

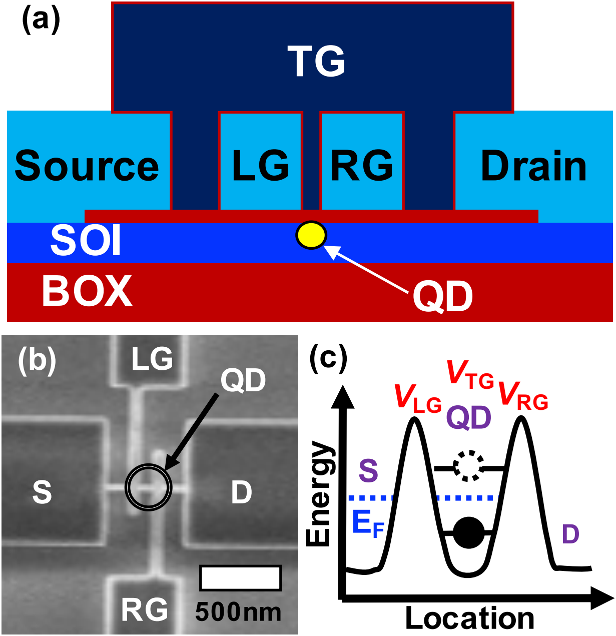
[44] Y. F. Lim, Y. Z. Xiong, N. Singh, R. Yang, Y. Jiang, D. S. H. Chan, W. Y. Loh, L. K. Bera, G. Q. Lo, N. Balasubramanian, and D. . Kwong, "Random telegraph signal noise in gate-all-around si-finfet with ultranarrow body," IEEE Elec. Dev. Lett. **27**, 765–768 (2006).

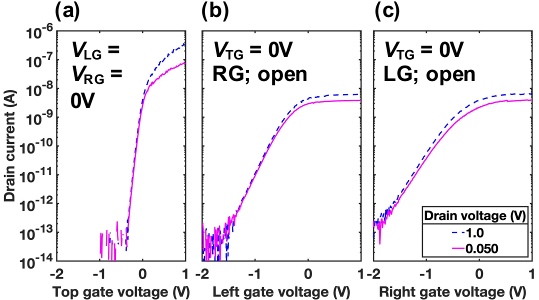
[45] S. Yang, K. H. Yeo, D. Kim, K. Seo, D. Park, G. Jin, K. Oh, and H. Shin, "Random telegraph noise in n-type and p-type silicon nanowire transistors," in *2008 IEEE International Electron Devices Meeting* (2008) pp. 1–4.

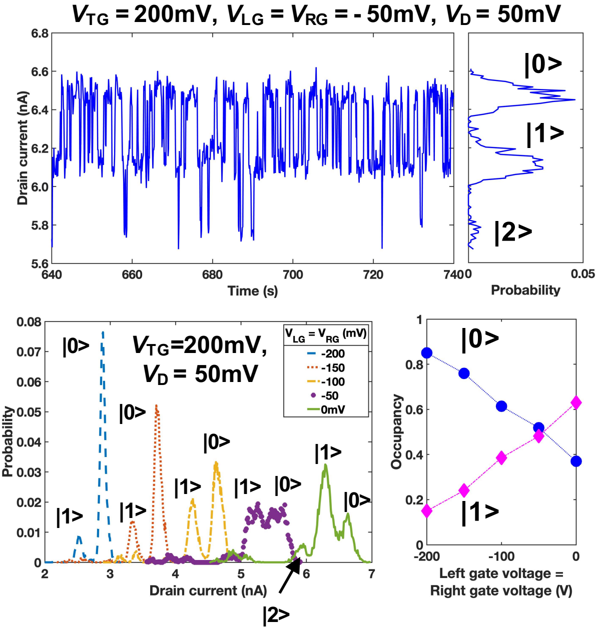
[46] G. Yamahata, K. Nishiguchi, and A. Fujiwara, "Gigahertz single-trap electron pumps in silicon." Nat. Com. **5**, 5038–5038 (2013).

[47] G. Yamahata, S. P. Giblin, M. Kataoka, T. Karasawa, and A. Fujiwara, "Gigahertz single-electron pumping in silicon with an accuracy better than 9.2 parts in 10^7," Appl. Phys. Lett. **109**, 013101 (2016).

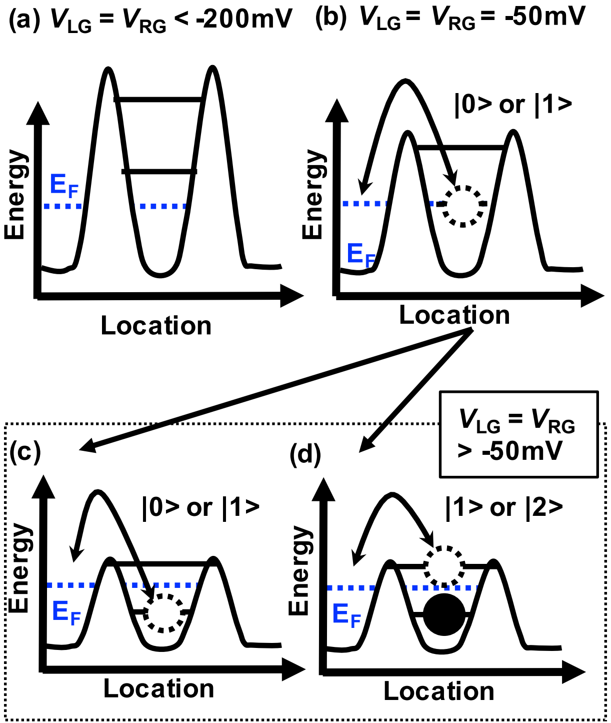
[48] K. Ibukuro, F. Liu, M. K. Husain, M. Sotto, J. W. Hillier, L. Zuo, I. Tomita, Y. Tsuchiya, H. N. Rutt, and S. Saito, "Dataset for silicon single-electron random number generator based on randomtelegraph signals at room temperature," `https://doi.org/10.5258/SOTON/D1488`, (University of Southampton, 2020).

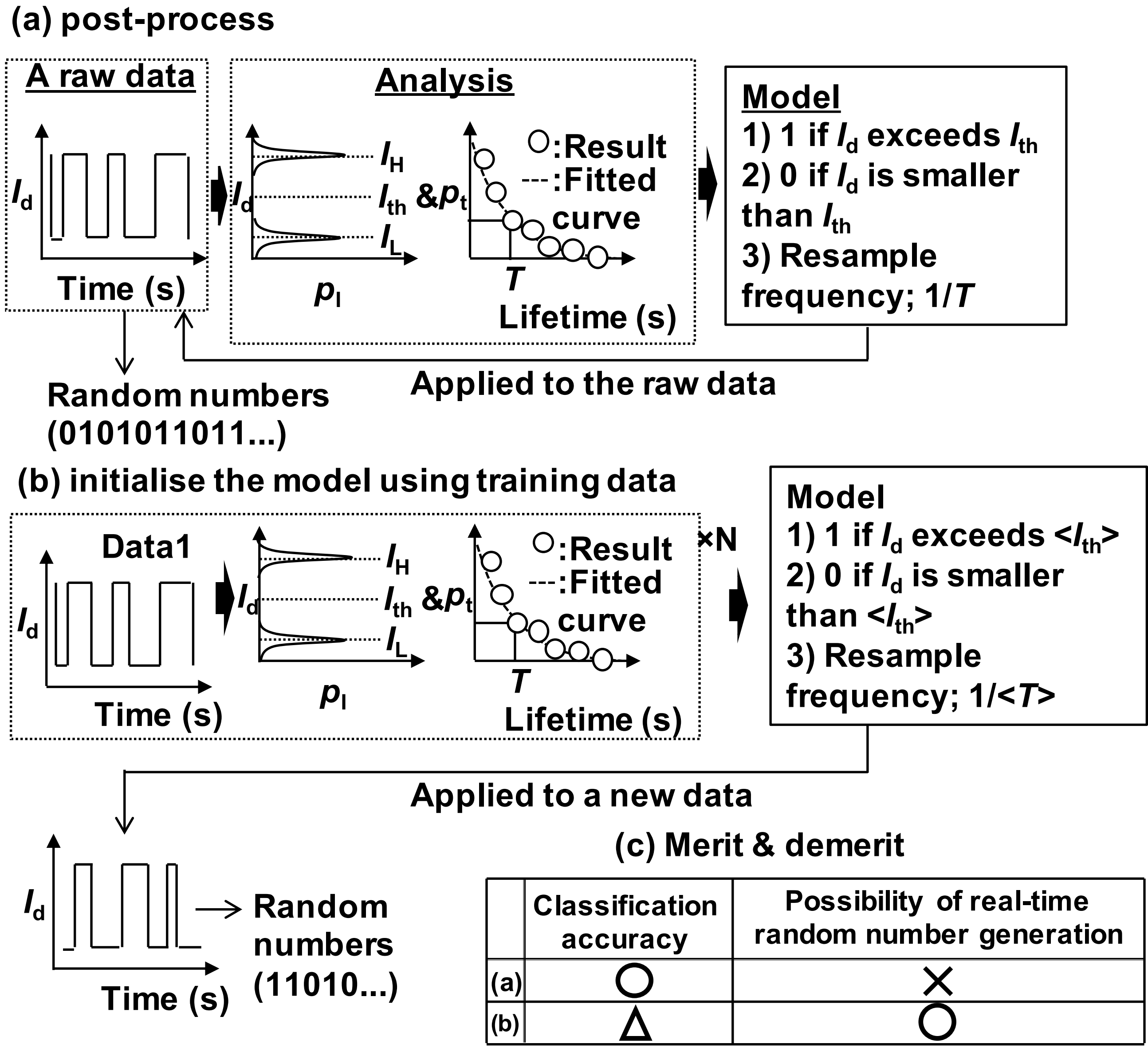

(a)
TG
Source
LG
RG
Drain
SOI
BOX
QD
(b)
LG
QD
S
D
RG
500nm
(c)
V_{TG}
V_{LG}
V_{RG}
QD
S
E_F
D
Energy
Location


(a)
(b)
(c)
V_{LG} =
V_{RG} =
0V
V_{TG} = 0V
RG; open
V_{TG} = 0V
LG; open
Drain current (A)
10^{-6}
10^{-7}
10^{-8}
10^{-9}
10^{-10}
10^{-11}
10^{-12}
10^{-13}
10^{-14}
-2
-1
0
1
Top gate voltage (V)
Left gate voltage (V)
Right gate voltage (V)
Drain voltage (V)
1.0
0.050



VTG = 200mV, VLG = VRG = - 50mV, VD = 50mV
Drain current (nA)
6.8
6.6
6.4
6.2
6.0
5.8
5.6
640
660
680
700
720
740
Time (s)
|0>
|1>
|2>
0
0.05
Probability
0.08
0.07
0.06
0.05
0.04
0.03
0.02
0.01
0
2
3
4
5
6
7
Drain current (nA)
Probability
VTG=200mV,
VD = 50mV
VLG = VRG (mV)
-200
-150
-100
-50
0mV
|0>
|1>
|0>
|1>
|0>
|1>
|1>
|0>
|1>
|0>
|2>
1
0.8
0.6
0.4
0.2
0
-200
-100
0
Occupancy
Left gate voltage =
Right gate voltage (V)
|0>
|1>


(a) $V_{LG} = V_{RG} < -200mV$
(b) $V_{LG} = V_{RG} = -50mV$
Energy
E_F
Location
|0> or |1>
Energy
E_F
Location
$V_{LG} = V_{RG}$ > -50mV
(c)
(d)
Energy
|0> or |1>
E_F
Location
Energy
|1> or |2>
E_F
Location



(a) post-process

A raw data

Analysis

Model
1) 1 if I_d exceeds I_{th}
2) 0 if I_d is smaller than I_{th}
3) Resample frequency; 1/T

Applied to the raw data

Random numbers (0101011011...)

(b) initialise the model using training data

Data1

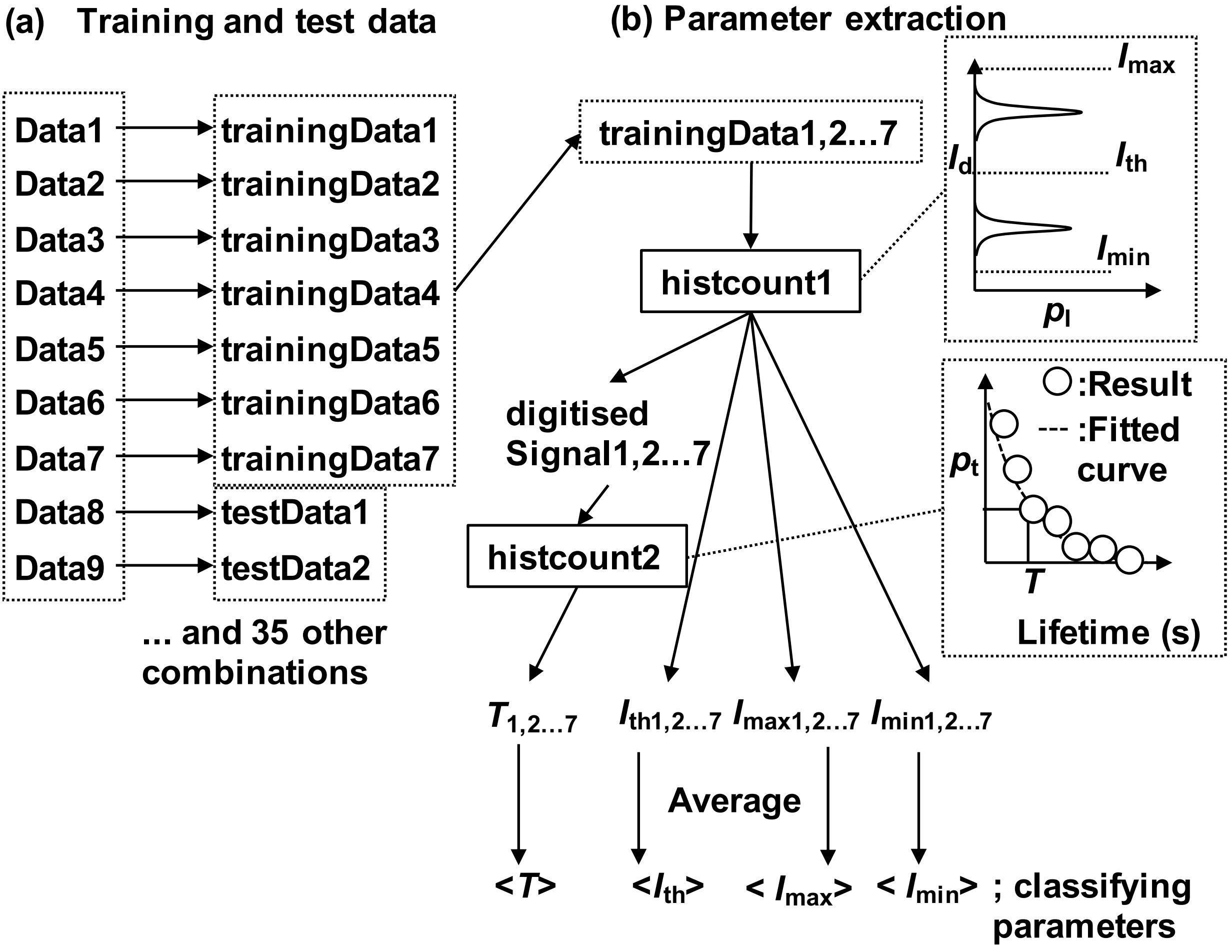
Model
1) 1 if I_d exceeds <I_{th}>
2) 0 if I_d is smaller than <I_{th}>
3) Resample frequency; 1/<T>

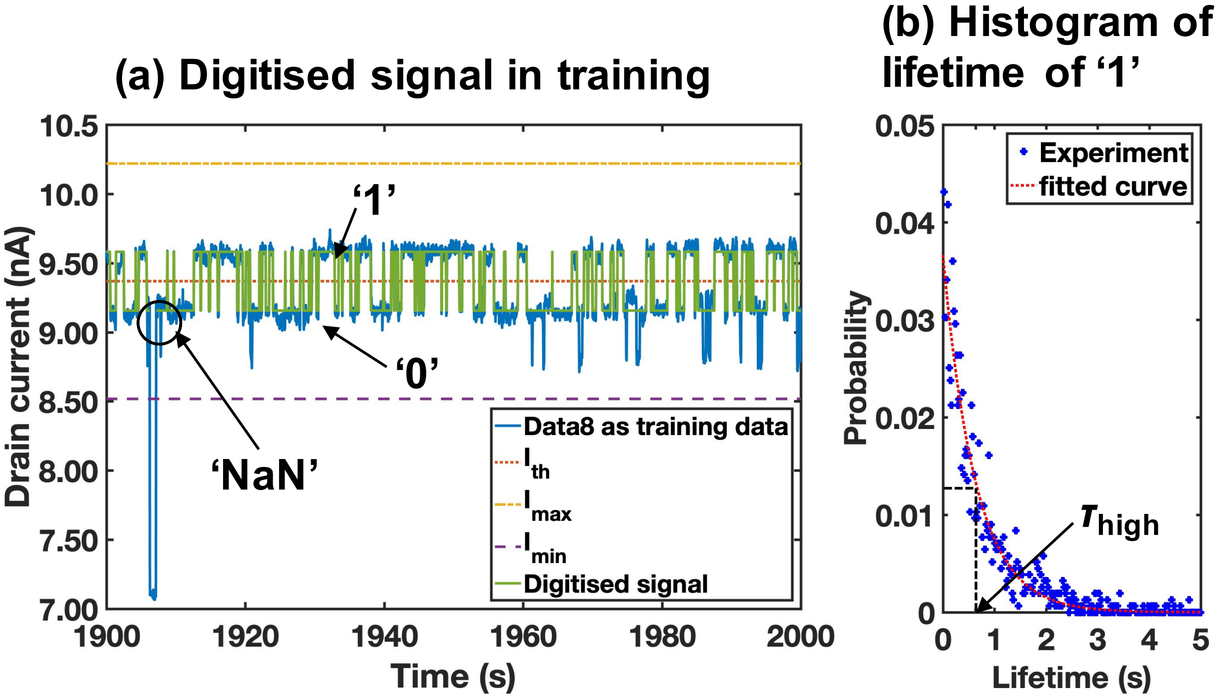
Applied to a new data

Random numbers (11010...)

(c) Merit & demerit

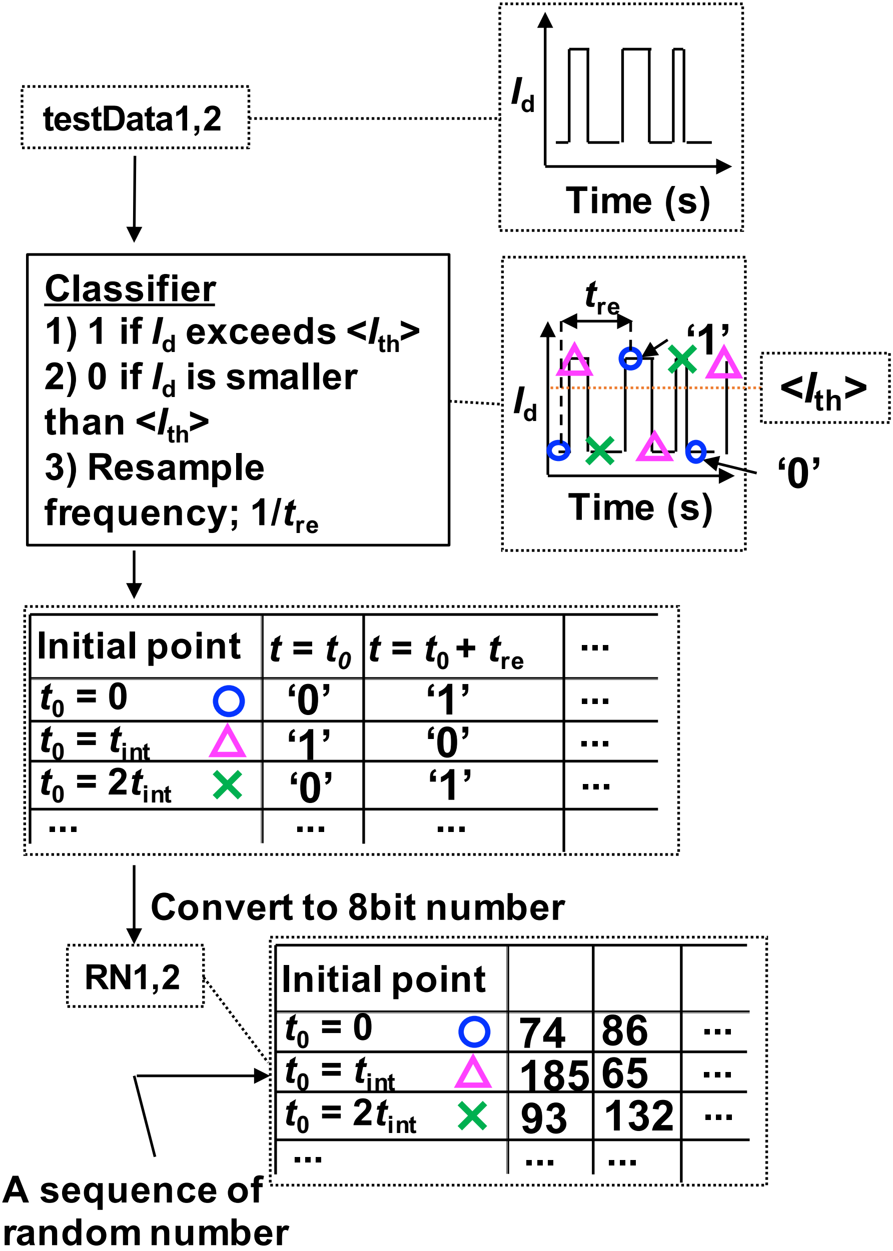
	Classification accuracy	Possibility of real-time random number generation
(a)	○	×
(b)	△	○


(a) Training and test data
Data1
Data2
Data3
Data4
Data5
Data6
Data7
Data8
Data9
trainingData1
trainingData2
trainingData3
trainingData4
trainingData5
trainingData6
trainingData7
testData1
testData2
… and 35 other combinations
(b) Parameter extraction
trainingData1,2…7
histcount1
digitised Signal1,2…7
histcount2
I_{max}
I_d
I_{th}
I_{min}
p_I
○ :Result
--- :Fitted curve
p_t
T
Lifetime (s)
$T_{1,2…7}$
$I_{th1,2…7}$
$I_{max1,2…7}$
$I_{min1,2…7}$
Average
$<T>$
$<I_{th}>$
$< I_{max}>$
$< I_{min}>$
; classifying parameters



(a) Digitised signal in training

(b) Histogram of lifetime of ‘1’



Classifier

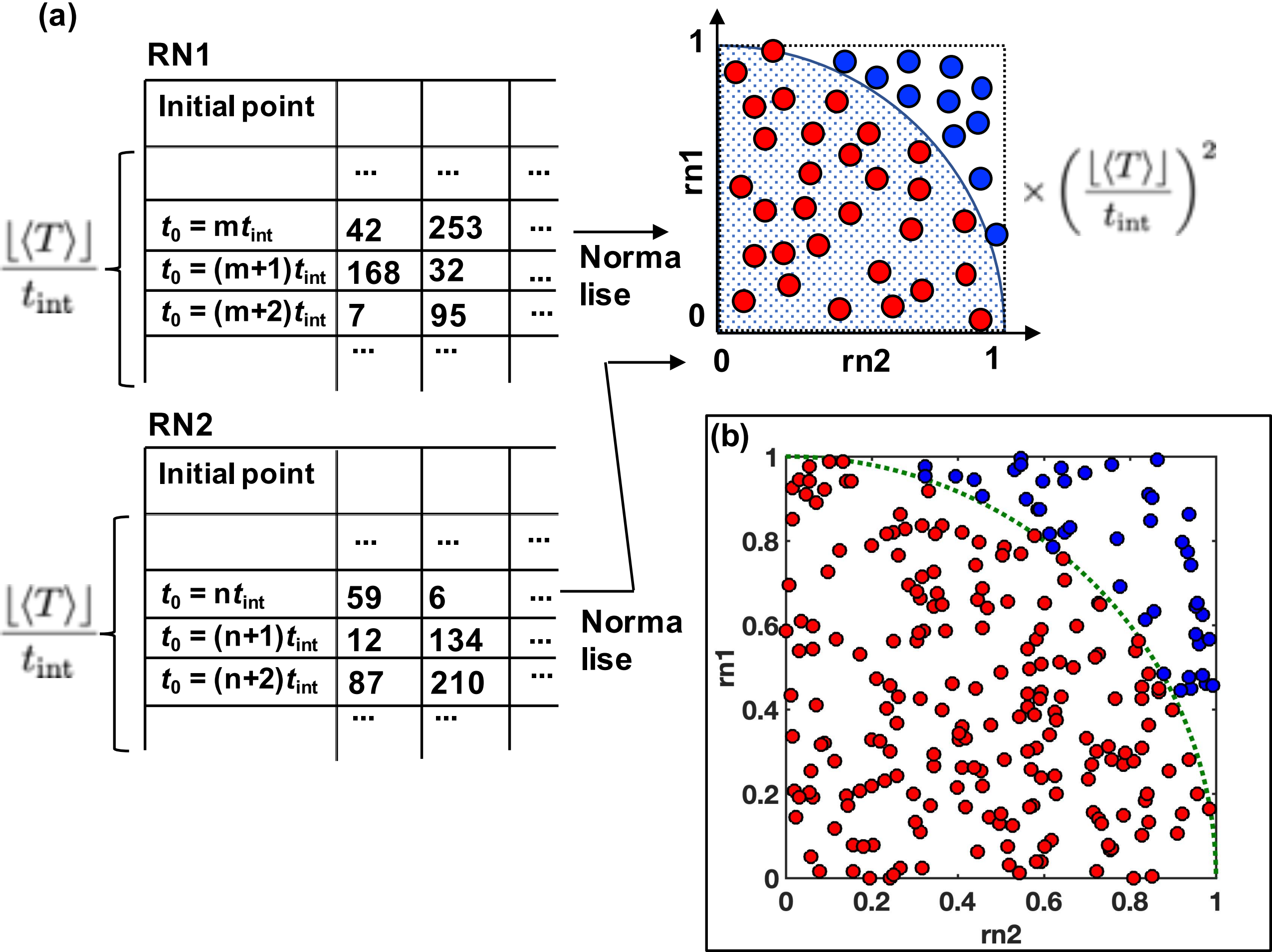
1) 1 if I_d exceeds $<I_{th}>$

2) 0 if I_d is smaller than $<I_{th}>$

3) Resample frequency; $1/t_{re}$

Initial point	$t = t_0$	$t = t_0 + t_{re}$	…
$t_0 = 0$ ○	'0'	'1'	…
$t_0 = t_{int}$ △	'1'	'0'	…
$t_0 = 2t_{int}$ ×	'0'	'1'	…
…	…	…	

Initial point			
$t_0 = 0$ ○	74	86	…
$t_0 = t_{int}$ △	185	65	…
$t_0 = 2t_{int}$ ×	93	132	…
…	…	…	

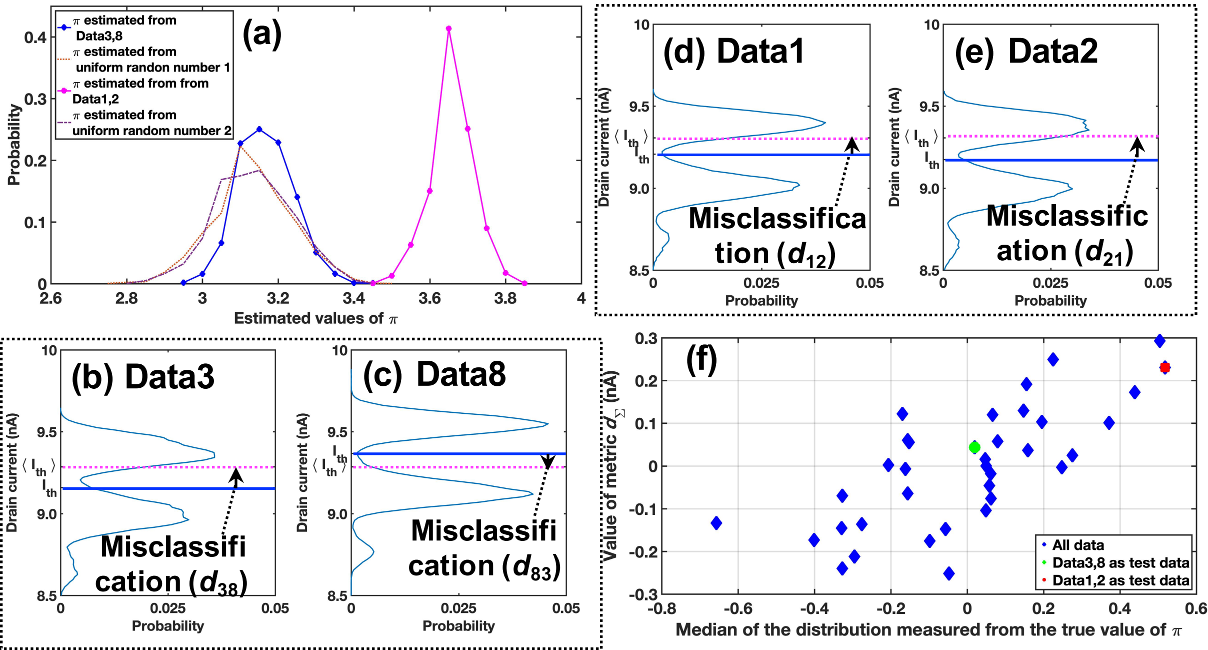

(a)
RN1
RN2
$\frac{\lfloor \langle T \rangle \rfloor}{t_{\rm int}}$
$\frac{\lfloor \langle T \rangle \rfloor}{t_{\rm int}}$
Norma lise
Norma lise
1
0
rn1
0
rn2
1
$\times \left(\frac{\lfloor \langle T \rangle \rfloor}{t_{\rm int}} \right)^2$
(b)
1
0.8
0.6
0.4
0.2
0
rn1
0
0.2
0.4
0.6
0.8
1
rn2

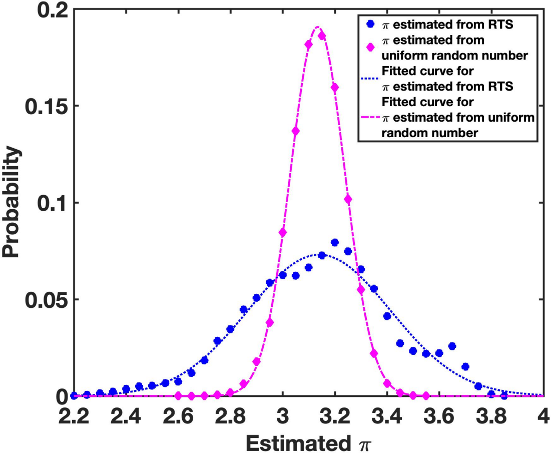
RN1

Initial point			
	···	···	···
t_0 = m$t_{\rm int}$	42	253	···
t_0 = (m+1)$t_{\rm int}$	168	32	···
t_0 = (m+2)$t_{\rm int}$	7	95	···
	···	···	

RN2

Initial point			
	···	···	···
t_0 = n$t_{\rm int}$	59	6	···
t_0 = (n+1)$t_{\rm int}$	12	134	···
t_0 = (n+2)$t_{\rm int}$	87	210	···
	···	···	


(a)
π estimated from Data3,8
π estimated from uniform randon number 1
π estimated from from Data1,2
π estimated from uniform random number 2
Probability
0
0.1
0.2
0.3
0.4
2.6
2.8
3
3.2
3.4
3.6
3.8
4
Estimated values of π
(d) Data1
(e) Data2
Drain current (nA)
10
9.5
9.0
8.5
$\langle I_{th} \rangle$
I_{th}
Misclassification (d_{12})
Misclassification (d_{21})
0
0.025
0.05
Probability
(b) Data3
(c) Data8
Misclassification (d_{38})
Misclassification (d_{83})
(f)
Value of metric d_{Σ} (nA)
0.3
0.2
0.1
0
-0.1
-0.2
-0.3
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
Median of the distribution measured from the true value of π
All data
Data3,8 as test data
Data1,2 as test data


0.2
0.15
0.1
0.05
0
Probability
2.2
2.4
2.6
2.8
3
3.2
3.4
3.6
3.8
4
Estimated π
● π estimated from RTS
◆ π estimated from uniform random number
Fitted curve for π estimated from RTS
Fitted curve for π estimated from uniform random number