Abstract—Network traffic characterization is required to obtain a good traffic model that can be used in the performance prediction of an intra-chip communication defined under multiples design parameters. In this paper we analyze the real traffic generated by several configurations of a MPSoC platform running the same multimedia MJPEG application, as we capture some statistical characteristics related to its temporal and long-memory behavior. Using some statistical techniques based on the autocorrelation function and spectrum analysis, we estimated the long-range dependence of on-chip traffic in order to explore how it changes with every platform configuration. Such traffic characterization shows also that packet injection distribution will be different regarding every configuration, but at the end keeping the total traffic on the chip almost unmodified.

Keywords: Autocorrelation Function, Hurst Parameter, Long-Range Dependence, MPEG-2 Decoder, NoC Performance, Packet Injection, Self-similarity, Traffic characterization.

I. INTRODUCTION

Modern system design methodologies for Multi Processor System on Chip (MPSoCs) include in their design steps like application modeling, performance evaluation and architectural synthesis. The first step consists in describing the applications that will run on the MPSoC platform. The second step starts with a general application-architecture mapping and uses it to evaluate the performance of the MPSoC in terms of throughput, latency and fifo (buffer) usage. The last step optimizes the on-chip computation and communication components in order to cope with the required performance.

As SoCs grow in complexity and size, one of these components, on-chip communication, is becoming increasingly important. Proposals of Networks on-chip (NoCs) for implementing communication in complex MPSoCs are justified by reusability, scalability and energy efficiency properties displayed by these networks [1].

As the area of NoC design is not yet mature, modest research work has been presented in the field of NoC traffic analysis and modeling. In a wide sense, network traffic modeling is a critical first step towards understanding and sorting out network power/performance related issues. Extensive prior research in the area of classic networks such as the Internet, Ethernet, and wireless LANs transporting TCP/IP, HTTP, and FTP traffic among others, has shown that predicting performance of a network is a difficult task. Stochastic modeling of the traffic must be used and it has been shown that this modeling must take into account second order stochastic properties (covariance in addition to marginal law) [2]. Second-order (or temporal) statistics of the autocorrelation function are the statistical properties that capture burstiness (or variability) in time series which characterize, for instance, traffic patterns in real networks [3]. In particular, the autocorrelation function, as a function of a time lag, decreases polynomially rather than exponentially. The existence of such nontrivial correlation “at a distance” is referred to as Long-Range Dependence (LRD).

Several works [4][5][6][7] have shown that on-chip traffic might require a similar stochastic modeling. They have found long-range dependent behavior in communications between different on-chip resources. The presence of such long-range dependence (LRD) in embedded applications, could imply that IPs communication behavior will not be correctly modeled by a simple random traffic even if it is adjusted to a particular Markovian processes, or to a traditional autoregressive moving average (ARMA) model that, so far, have been mostly used in system-level analyses [8][9].

On the other hand, several authors have used synthetic workloads to validate some NoC features, without concerning for which applications will be finally running on these systems, generating some non real-life traffic that cannot be used to drive a realistic network design-space exploration [9][10].

Thus, it is necessary to analyze the on-chip traffic generated by the current application in order to identify the presence of long-memory dependence in the system and how it will be affecting the network performance in terms of throughput, latency and buffer utilization. By our knowledge no work has been presented on characterizing the degree of long-range dependence in the traffic of a multimedia application realized in a MPSoC platform, considering spatial variations in its architecture and temporal variations in the on-chip traffic. The objective of this work is to perform such a characterization using techniques based on the Hurst parameter estimation and self-similarity measurement.

The paper is organized as follow: section II exposes the related work. Section III briefly explains the concept of traffic analysis showing its importance for network performance. Section IV introduces some topics and definitions related long-
range dependence. Section V exposes the case study used for our simulations and traffic analysis, explaining as the chosen multimedia application and the MPSoC platform explored. Section VI summaries our analysis and experimental results about traffic characterization. Finally section VII concludes our paper.

II. RELATED WORK

Varatkar and Marculescu [4] have presented a seminal paper where they have shown that the aggregated throughput of the macro-block arrival process at the IDCT/IQ module of a hardware MPEG-2 decoder exhibits long-range dependence. Considering that their experiments were not done at cycle-accurate level, they capture traffic characteristics between pairwise nodes rather than for the entire network, ignoring the spatial dimension of NoC traffic and how the LRD impact the network-on-chip. Tedesco et al. [20] compare NoC performance, in terms of throughput and latency, when different traffic modeling methods are used for multimedia applications. They also consider the interfering traffic generated by every IPs within the SoC, but without any analysis about the temporal dependence of burstiness observed in their models, and its relation with the system architecture or topology. Bogdan and Marculescu [7] have investigated the origin of self-similarity1 and long-range dependence properties in network traffic based on the existence of energy levels and packet injection rate parameters in NoCs. They have presented a statistical physics inspired framework which can capture and reveal important characteristics [5]. In the case of traffic, the analytic approach to traffic modeling uses mathematical functions to characterize some chosen parameters of the workloads generated by various classes of applications. For example, starting from real multimedia traces, one can build an analytical model that captures the long-range dependencies, so that various performance and cost metrics such as packet loss probability and buffer size can be optimized.

In this work we explore the long-range dependent behavior, within the on-chip traffic, related to several spatial modifications on the resources compounding the MPSoC platform, while keeping unchanged network parameters such as topology, routing protocol, switching technique and others. By analyzing multiple traffic traces captured from several simulation of our case study, we could identify the packet injection arrival processes obtained from every network resource, used subsequently to estimate the Hurst Parameter on each case.

IV. LONG-RANGE DEPENDENCE

A. Definition

Let \( X(t) \), \( t \in \mathbb{Z}^+ \), be a stochastic process denoting the traffic volume (bits, packets, etc.) at time instance \( t \). The mean \( \mu = \mathbb{E}[X] \), the second-order statistics such as variance \( \mathbb{E}[(X - \mathbb{E}[X])^2] \), and the autocorrelation function \( r(k) \), reveal important information about the time series that capture the actual traffic characteristics [5].

Long-range dependence (LRD) is a property of a stochastic process \( X(t) \) defined as a slow decrease of its autocorrelation function (ACF) [12]. The “dependence” part means that successive time events are not independent of each other. On the contrary, they are indeed dependent. The “long-range” part means that the dependence is not only among adjacent events. But events that are very far apart may be correlated with each other too. However, correlation does actually taper off with distance. The point is that it decays very slowly. If the correlation decays exponentially with distance, this is short-range dependence (SRD). But if it decays according to a power law, we have LRD.

More formally, LRD is actually defined as having an ACF that falls off polynomially, with an exponent smaller than 1:

\[ r(k) = \mathbb{E}[X(t)X(t+k)] = \frac{\alpha}{|k|^{\beta}} \]

1 In mathematics, a self-similar object is exactly or approximately similar to a part of itself (i.e. the whole has the same shape as one or more of the parts). Self-similarity is a typical property of fractals [11]. In the case of traffic modeling, stochastic self-similarity adds to the traffic a probabilistic behavior. If we consider the time series which may characterize some real-data traces, it may be possible to expect an approximate similarity with respect to the shape of the autocorrelation function [11].
$$r(k) \approx k^{-\beta} \quad 0 < \beta < 1$$

This function does decay as the time lag $k$, between the random variables, grows. There is a stronger correlation across short distances than across long distances, but the correlation across long distances is still strong enough to make a significant contribution to the sum. As a result, the ACF is non-summable:

$$\sum_{k=1}^{\infty} r(k) = \infty$$

Consequently, LRD reflects the ability of the process to be highly correlated with its past, because even at large lags, the ACF is not negligible. This property is also linked to self-similarity which is more general: it can be shown that asymptotic second order self-similarity implies LRD [11].

In the ACF defined above, the exponent $\beta$ (also called scaling index) provides a parameter to tell how much a process is long-range dependent ($0 < \beta \leq 1$). The Hurst exponent, noted $H$, is the classical parameter for describing self-similarity. Because of the analogy between LRD and self-similarity, it can be shown that a simple relation exists between $H$ and $\beta$: $H = (2 - \beta)/2$. As a consequence, $H$ (1/2 $< H < 1$) is the commonly used parameter for LRD. Note that when $H = 0.5$, there is no LRD (this is also referred to as SRD).

B. Estimation of the Hurst Parameter

There are several tests to measure $H$, as LRD manifests itself in a number of ways. One such method uses time-domain analysis based on the re-scaled adjusted range statistic, denoted as the R/S statistic [12].

Given observations $(X_k; k = 1, 2, \ldots, n)$ (with sample mean $\bar{X}(n)$ and sample variance $S^2(n)$), the rescaled adjusted range statistics is given by

$$\frac{R(n)}{S(n)} = \frac{1}{S(n)} \left[ \max(0, W_1, W_2, \ldots, W_n) - \min(0, W_1, \ldots, W_n) \right]$$

where $W_k = (X_1 + X_2 + \ldots + X_k) - k\bar{X}(n)$, $1 \leq k \leq n$.

In his study of the rescaled adjusted range [13], Hurst found that many historical records appeared to well be represented by

$$E \left[ \frac{R(n)}{S(n)} \right] \sim cn^{H}, \quad \text{as } n \to \infty$$

with Hurst parameter about 0.7. On the other hand, if the $X_i$’s are Gaussian pure noise or short-range dependent, then $H = 0.5$ in (4) and the discrepancy is referred to as the Hurst effect. The Hurst effect is fully accounted for by stationary stochastic processes with long-range dependence. To obtain $H$, one plots $\log_{10}(\frac{R(n)}{S(n)})$ versus $\log_{10}(n)$. This is called an R/S log-log plot, where the slope of the R/S line is $H$. This slope is calculated using an inverse-variance weighted least-squares curve fit.

C. Implications of LRD in NoC performance

The effect of the high variability that is associated with self-similarity and LRD has been studied mostly in the context of communication networks. Communication networks have often been analyzed using Poisson-related models of traffic, which indicated that the variance in load should smooth out over time and when multiple data sources are combined. This property allows for precise evaluations of how much buffer space is needed to support communications at a given average rate, and for the design of algorithms for provisioning different levels of service to different users in a macro-network.

Conversely, using LRD workload models that capture the burstiness of network traffic leads to different results in performance evaluations. For example, such evaluations lead to larger and more realistic estimates of required buffer space and other parameters. These models also extend a shadow of doubt on the feasibility of various schemes for providing guaranteed levels of service (QoS), because the high variability of the load prevents the system from being able to estimate how much extra capacity is available.

The most striking feature of LRD traffic is that burstiness is displayed across several time scales. This burstiness can drive queueing systems into overflow state violating the steady state assumption that is the basis of queueing network models and many other types of analysis based on Poisson processes [14]. Although significant multiplexing gain can be achieved for LRD traffic streams, the burstier stream will dominate the queueing tail distribution. This means that a traffic stream with lower Hurst parameter may suffer the same mean delay as a traffic stream with higher Hurst parameters.

As a consequence, for macro-networks as well as for on-chip networks, LRD should be taken into account if it is found in the traffic that the network will have to deal with. So, knowing the Hurst parameter helps the designer to choose the minimal buffer size for every router at each tile of a network, which, in turn ensures a certain QoS for running a multimedia application.

V. Study Case

A. The MPEG-2 Video Decoder

The application used for our experiments is a MJPEG decoder, which reads a stream of JPEG images (motion-JPEG), from an input peripheral and writes pixels into an output random access memory digital-to-analog converter (RAMDAC) [15]. The general structure of MJPEG decoding is shown in Figure 1.

![Figure 1: MJPEG decoder](image-url)
stream of coefficients. IQ performs the inverse quantization. IDCT performs the inverse discrete cosine transform. And LIBU is not a JPEG operation, but it is necessary to adapt the pixel stream to a given output controller. The MJPEG decoder constraints are: 256-level grayscale images; frequency of 50 MHz and 25 frames/s for 128 x 128 images.

The figure 2 shows a Task Communications Graph (TCG), which defines the coarse grain parallelism of the software application implemented in the MPSoC platform. Such coarse-grained parallelization is done two ways: first, the video stream is split at image boundaries and distributed among different decoder pipelines where each decoder pipeline is split in the successive tasks.

![Figure 2: Task Communication Graph of MJPEG decoder](image)

B. MPSoC Simulation Environment

For the implementation of the above application, we have used an open source, SystemC-based, cycle-accurate and bit-accurate simulation environment known as SoClib [16]. The environment contains cycle-accurate models for various IPs. For instance, it contains a MIPS R3000 processor with its associated data and instruction cache, on-chip memories, a display component (TTY), and other components such as input/output interfaces.

![Figure 3: SoC Platform Configuration No. 10](image)

The application running on the MIPS is composed, in addition to bootstrapping information, of the C program cross-compiled with GCC to a MIPS target. The network-on-chip selected is the DSPIN (Distributed Programmable Integrated Network) developed as an evolution of the SPIN network [17]. It uses a mesh topology, static XY routing and wormhole memorization in the switches.

Thus, the hardware of the resulting MPSoC is composed of a global interconnect, local clusters, with some processors and RAM in which it is implemented each image decoding pipeline, a local cluster containing image input stream coprocessor, and a local cluster containing image output frame-buffer coprocessor. The figure 3 shows one of our 2x3 mesh architecture configuration used to analyze the traffic on the network.

The simulations were done at cycle-accurate level in order to observe the exact behavior of what would occur on a real chip. For this purpose, we set up ten different configurations to observe the transactions between every IP resource and the network while the system was running two MJPEG decoder pipelines, following several TCG’s tasks mappings on every resource within each configuration, as we show in the table 1.

While the application code was running on the whole system, we record the sequence number and timing of flits generated by each IP resource in a trace file, which was used afterwards to specify the time series of the packet injection arrival processes ingoing to the network.

<table>
<thead>
<tr>
<th>Some Analized Configurations</th>
<th>R00</th>
<th>R01</th>
<th>R02</th>
<th>R10</th>
<th>R11</th>
<th>R12</th>
</tr>
</thead>
<tbody>
<tr>
<td>config2</td>
<td>TG</td>
<td>RAM</td>
<td>MIPS2 (PIPE0)</td>
<td>MIPS3 (PIPE3)</td>
<td>MIPS4 (PIPE1)</td>
<td>TTY RAMDAC</td>
</tr>
<tr>
<td>config4</td>
<td>TG</td>
<td>MIPS1 (PIPE0)</td>
<td>MIPS2 (PIPE0)</td>
<td>MIPS3 (PIPE3)</td>
<td>RAM</td>
<td>MIPS5 (PIPE1)</td>
</tr>
<tr>
<td>config6</td>
<td>TG</td>
<td>RAM</td>
<td>TTY RAMDAC</td>
<td>MIPS3 (PIPE3)</td>
<td>MIPS4 (PIPE1)</td>
<td>MIPS5 (PIPE1)</td>
</tr>
<tr>
<td>config8</td>
<td>MIPS5 (PIPE0)</td>
<td>TG</td>
<td>TTY RAMDAC</td>
<td>RAM</td>
<td>MIPS4 (PIPE1)</td>
<td>MIPS5 (PIPE1)</td>
</tr>
<tr>
<td>config10</td>
<td>RAM</td>
<td>MIPS1 (PIPE0)</td>
<td>TTY RAMDAC</td>
<td>MIPS3 (PIPE1)</td>
<td>TTY RAMDAC</td>
<td>MIPS5 (PIPE1)</td>
</tr>
</tbody>
</table>

Table 1: Some Analized Configurations

VI. EXPERIMENTAL RESULTS

A. Traffic evaluation according to position on NoC

The figure 4(a) shows the traffic generated in the configuration config2 by the processor called MIPS3 and connected to router R10, while it was running the pipeline0 (PIPE0) of a MJPEG decoder. The ACF of this traffic is show in figure 4(b).

![Figure 4: (a)Traffic trace for PIPE0 on config2. (b) Its ACF.](image)

As can be seen in this figure, at small values of time lag the autocorrelation seems to have a slow decay, possibly motivated by a SRD effect on such time interval, but at long values of time lag, such autocorrelation shows not negligible values that certainly will imply a LRD behavior, how it will be verified more bellow.

Simulating another configuration, the figure 5(a) shows the traffic generated by a MIPS running the same task PIPE0, but now allocated in a different tile within the network. The
The respective ACF of this time series is shown in figure 5(b). Because this ACF decays more slowly than that of figure 4(b), it is more likely that such traffic possesses a higher LRD effect, while keeping imperceptible any SRD effect.

These both cases are just an example of how the traffic and its LRD behavior can be different for the task application running on a IP core allocated in different position tiles along the network.

B. Traffic Injection Distribution

Traffic injection distribution models the ratio of total network traffic each router injects into the network and their distribution.

C. Hurst Parameter Estimation

A number of techniques have been proposed to estimate the Hurst parameter. Some of the most popular were shown on section IV-(B) such as the R/S statistic and the analysis of variances of the aggregated processes. Although there are many other techniques, over time, the wavelet-based methods have acquired popularity in estimating LRD traffic [18].

By using the R free software environment for statistical computing and graphics [19], we have calculated the Hurst parameter for every traffic time series, obtained in our simulations, using three methods: R/S statistic, a wavelet-based and a periodogram-based method.

The table IV shows the calculated values for the traffic time series captured from RAM memory in every configuration. In this table, the values from the R/S method show the high long-range effect of every time series, keeping almost invariable on each configuration. On the other hand, the values from the wavelet-based and periodogram-based methods are more vulnerable to SRD effects also found on these time series. In a wide sense, those values prove that the Hurst parameter for the packet arrival process generated from the RAM memory remain independent of its position on the network. The figure 7 shows the periodogram log plot used to calculate the Hurst parameter on the traffic generated on configurations config2 and config10.
The table V and VI shows the calculated values for the traffic time series captured from processors running the PIPE0 and PIPE1 tasks of the MJPEG decoder pipelines, on every simulated configuration. In these tables, the values from the R/S method show the long-range effect on each time series, changing with each configuration.

### Table V: $H$ Estimation for traffic generated by PIPE0 on each configuration

<table>
<thead>
<tr>
<th>PIPE0</th>
<th>RsFit</th>
<th>WaveletFit</th>
<th>PerFit</th>
</tr>
</thead>
<tbody>
<tr>
<td>config2_r10</td>
<td>0.784</td>
<td>0.398</td>
<td>0.405</td>
</tr>
<tr>
<td>config10_r10</td>
<td>0.719</td>
<td>0.725</td>
<td>0.749</td>
</tr>
<tr>
<td>config4_r11</td>
<td>0.722</td>
<td>0.451</td>
<td>0.522</td>
</tr>
<tr>
<td>config6_r10</td>
<td>0.755</td>
<td>0.443</td>
<td>0.464</td>
</tr>
<tr>
<td>config8_r10</td>
<td>0.733</td>
<td>0.447</td>
<td>0.419</td>
</tr>
</tbody>
</table>

### Table VI: $H$ Estimation for traffic generated by PIPE1 on each configuration

<table>
<thead>
<tr>
<th>PIPE1</th>
<th>RsFit</th>
<th>WaveletFit</th>
<th>PerFit</th>
</tr>
</thead>
<tbody>
<tr>
<td>config2_r11</td>
<td>0.815</td>
<td>0.407</td>
<td>0.470</td>
</tr>
<tr>
<td>config10_r12</td>
<td>0.954</td>
<td>0.364</td>
<td>0.435</td>
</tr>
<tr>
<td>config4_r10</td>
<td>0.827</td>
<td>0.718</td>
<td>0.722</td>
</tr>
<tr>
<td>config6_r12</td>
<td>0.801</td>
<td>0.389</td>
<td>0.462</td>
</tr>
<tr>
<td>config8_r12</td>
<td>0.891</td>
<td>0.402</td>
<td>0.463</td>
</tr>
</tbody>
</table>

Figure 8: (a) $H$ estimation for PIPE0’s traffic on config2. (b) $H$ estimation for PIPE0’s traffic on config10.

Such figure 8 shows the periodogram log plot calculating the Hurst parameter for the PIPE0’s traffic on configurations config2 and config10. The last one shows a big LRD effect of the packet arrival process generated from the MIPS3 processor when connected to router R10 within such configuration.

### VII. CONCLUSIONS AND FUTURE WORK

In this paper we characterized the real traffic generated by several configurations of a MPSoC platform running the same multimedia MJPEG application, but with different resources spatial distribution, as we captured statistical characteristics related to long-range dependent behavior. Such traffic characterization showed that the long-range behavior can be different for the traffic generated by every resource depending on where it is allocated in the network architecture, which must be considered when modeling the whole on-chip traffic. Our future research work will be on this direction, specifying new real traffic models based on LRD characterization for estimating a reliable on-chip network performance prediction.

**ACKNOWLEDGMENT**

This work has been partially supported by CAPES.

**REFERENCES**

[17] I. Miro-Panades, A. Greiner, and A. Sheibanyrad, “A Low Cost Network-on-Chip with Guaranteed Service Well Suited to the GALs Approach,” 1st Int. Conf. on Nano-Networks and Workshops (NanoNet-2006).