Abstract—Wireless Sensor Networks (WSNs) have recently enjoyed a tremendous rise in popularity. The current WSN node offerings, however, need both increased processing power and lower energy consumption in order to enable the full potential of such networks. To address these requirements, we explore the benefits of an innovative platform which combines a standard wireless node with very low cost reconfigurable hardware. In order to evaluate the efficiency of this pioneering approach three different networking and security protocols have been implemented on the present system: a) Turbo coding, b) Blowfish encryption and c) XMesh routing. Our real-world experiments demonstrate that our prototype system provides comparable performance to the existing microcontroller-based schemes (while in its productized version it could potentially be much faster) whereas, and more importantly, its overall energy consumption is from 70% to 93% lower than that triggered when a very widely used commercial WSN node is executing the exact same processing tasks.

I. INTRODUCTION

Wireless Sensor Networks (WSNs) have been used in various applications, including environmental monitoring ([11], [2]), military systems ([3], [4]), health care ([5], [6]) and industry automation ([7], [8]). Typically, nodes collect data from their sensors and, then, they process and transmit them in a periodic or event-driven manner. Data processing together with the wireless communication account for the greatest part of the energy consumed by a node. Commonly, the energy required for the radio transmission is significantly more than the energy spent for the computation tasks and, thus, many WSN designers have focused on improving the efficiency of the wireless communication procedure. However, due to the emerging WSN requirements, the energy spent on the computational tasks is expected to represent a much greater percentage of the overall WSN energy consumption in the near future. For example, most of the proposed techniques that are improving the communication efficiency - such as data aggregation, data compression and error control schemes - consume significant energy when executing their processing tasks. More importantly, as WSNs are engaged in new applications, the functions executed on the motes are becoming more complex and demanding (e.g. image/video processing and encryption/authentication).

Moving to a different sector, Field Programmable Gate Array (FPGA) manufacturers offer small Complex Programmable Logic Devices (CPLDs) which are programmable devices combining the architectural features of Programmable Logic Arrays (PLAs) with those of FPGAs. Their main characteristics are the low energy consumption coupled nowadays with a relatively high clock frequency when executing certain data manipulation tasks. CPLDs typically cost from $2-10.

In this paper, we propose a novel WSN node platform, which combines a widely used general purpose mote (CrossBow's MiCa2/IRIS [9]) with a Xilinx CoolRunner II CPLD device ([10]). By implementing the computationally intensive tasks in hardware, the energy consumption can be significantly reduced, while the use of reconfigurable devices offers the flexibility required by a general purpose WSN node. The small size and low cost of the CPLD make the proposed solution attractive from an integration perspective as well. To explore the potential of the platform, we present the actual speed and energy consumption when three interesting WSN related tasks are executed. The first scheme aims at improving the efficiency of the wireless communication by utilizing a Forward Error Correction mechanism (Turbo Code). Secondly an efficient routing scheme (XMesh) has also been implemented on this same prototype. Finally, the proposed scheme is executing an encryption/decryption protocol (Blowfish). As our real-world measurements clearly demonstrate by implementing these tasks (or their most computationally intensive kernels) in the proposed CPLD device, significant energy savings are achieved when compared with the existing microcontroller-based WSN nodes; in particular those energy savings can be from 70 up to 93% meaning that the introduction of the low-cost CPLD can increase the so important life-time of a battery operated node from seven to ten times.

II. RELATED WORK

Recently, a number of WSN platforms that incorporate reconfigurable devices have been introduced. In this section, we present those platforms and discuss their main differences when compared with our novel WSN node.

A. mPlatform

The mPlatform [11], introduced by Microsoft, is a "reconfigurable modular sensornet platform that enables real-time processing on multiple heterogeneous processors". The heart of the mPlatform is a scalable high-performance communication bus controlled by a low power CPLD device. The latter connects the different stackable modules of a node.
with the central processor and allows the time-critical data to be transmitted from one module to another with very low delay. The mPlatform can support a real-time sound source localization application utilizing four different acoustic channels, Fast Fourier Transform and sound classification; this application cannot be supported by the current motes which are build around traditional low-power buses such as I2C.

B. Atific Helicopter High Performance Multi-Radio WSN Platform

The design and implementation of an innovative high performance multi-radio WSN platform is presented in [12]. This platform includes both multiprocessor devices and extremely flexible and fully reconfigurable FPGAs. The use of four parallel radio transceivers with selectable frequency channels allows the development of communication protocols with high interference tolerance, low latency and high mesh-networking performance. Purely simultaneous data exchanges with several neighbours are possible in the proposed platform. The efficiency of their approach is demonstrated in two distinct applications which achieve ultra low latency and high network throughput.

The main differences between those systems and our approach lie at the actual use of the reconfigurable device on the mote as well as the targeted applications. In the case of the mPlatform, the reconfigurable device is utilized only in the intercommunication of the different stackable modules. In contrast, in our novel system the reconfigurable device executes certain data processing and networking tasks. On the other hand, the Atific platform aims at specific high-performance applications without severe energy restrictions (i.e. it is designed to be mains-powered rather than battery-powered). Our flexible, general purpose, yet innovative platform can be used in a far greater range of applications, albeit not as demanding as the ones implemented in the Atific platform; in application domains we target energy consumption is crucial and this is the principal motivation in adopting a reconfigurable device rather than simply using general purpose CPUs.

III. RESENSE Platform

A. Hardware

The sensor nodes utilized in our novel platform are probably the most widely used such nodes worldwide; those are Crossbow Technology’s ‘MICAz’ and ‘IRIS’ [9] motes. They include IEEE 802.15.4 compliant, ZigBee ready radio frequency transceivers integrated with an ATmega128-series micro-controllers. The motes are connected to Crossbow MDA100 sensor and data acquisition boards which provide the required sensors and a general prototyping area. We expanded these nodes by connecting them to a CPLD. The main reasons for selecting CPLDs over other solutions (FPGAs or ASICs) are their inheritable low energy consumption and low cost. When compared to ASIC devices, CPLDs cannot clearly match their low energy consumption and high processing power. However, using an ASIC in a general-purpose platform limits its reach to a predefined (i.e. at design time) set of applications. Furthermore, it is not a cost-effective solution, if the volumes are low while it requires a longer time to market. The desirable flexibility offered by CPLD devices, can also be provided by FPGAs (or coarse grain reconfigurable ASICs). FPGAs have significantly higher performance than CPLDs and their available hardware resources can accommodate much larger designs. Nonetheless, CPLDs are smaller in physical dimensions, much cheaper, and above all, more power efficient. Another important advantage of the CPLDs, is that they can retain their configuration data even when powered-off, thus no external memories are needed and the energy consumed on power up is negligible.

Our pioneering platform utilizes the CPLD Digilent X-Board [13] development platform which is connected to the IRIS and MicaZ motes. The actual CPLD is a very low-cost 256 macro-cell Xilinx CoolRunner-II CPLD [10]. The reconfigurable device is connected to a host (i.e. a workstation PC) through a USB 2.0 port for initial device configuration purposes and to a mote through the prototype area of the MDA100 board. One of the most important issues when implementing our prototype platform was the connection between the motes and the reconfigurable device. On the mote side, only 24 out of the 102 prototyping area pins are actually available since the remaining pins are either open or dedicated to a specific operation of the mote’s main micro-controller. Therefore, for the interconnection we employed a 16-lane data bus (8 lanes per direction) combined with 8 input/output control signals; those signals are connected to the JTAG pins of the CPLD. Figure 1 demonstrates our prototype wireless platform.

B. Software

For the development of our applications we have utilized the NesC programming language and TinyOS [14]; the latter is an operating system designed especially for embedded devices used in distributed WSNs. The CPLD designs were modeled in Verilog and in VHDL hardware description languages. Xilinx ISE 10.1 has been used for the synthesis and the actual design implementation.

C. Mote-CPLD Communication Protocol and Prototype Platform Limitations

On the top of our intercommunication infrastructure described in the last section we implemented a Digital I/O communication protocol to support the data transfers between the motes’ micro-controllers and the CPLD. The low-frequency I/O pins of the ATmega128-series micro-controllers, utilized in the MICAz/IRIS motes, restrained us from implementing a high-speed serial communication solution and, as a result, a parallel I/O protocol was adopted. Because of the limitations of the available pins mentioned in the previous subsection, we used a 16-bit data bus (8 bits per direction) along with a number of control lines. On the top of them a simple toggle (handshake) synchronization protocol is applied in order to guarantee the reliable data exchange between the devices.
There are two significant limitations of our intercommunication scheme. The first one has already been mentioned and it is related to the number of available I/O pins; the relatively low-number of available pins pose an upper bound on the data lines that can be used and therefore, constrain the intercommunication bandwidth that can be achieved. The second issue concerns the specific available I/O pins that can be used for the intercommunication scheme in both our MICAZ and IRIS development systems. The ATMega128-series micro-controllers, utilized in those motes, have a number of 8-bit I/O ports. The data on each port can be accessed either as an 8-bit word or each pin (bit) can be read independently.

The available pins on our prototyping platform cannot be grouped in ports and, as a result, the micro-controller has to read the incoming information and to write the outgoing data in a bit-by-bit fashion. This leads to significant latencies and therefore performance loss (which is highlighted in our performance results demonstrated on the following sections). It should be stressed that these problems are expected to be solved in the final production platform. Our custom-designed board will include both the ATMmega micro-controller and the CPLD and it will both increase the available bandwidth between the two devices (through the support of wider buses) and decrease the intercommunication delay (by the proper grouping of the data lines of the micro-controller).

IV. PLATFORM EVALUATION

The prototype system was evaluated based on two metrics: execution time and energy consumption. Both are significant parameters in WSNs, since it is certainly desirable to increase the limited processing power of the node while also increasing its lifetime by lowering the energy consumption. Our performance results are based on real-world experiments in which a mixed-signal oscilloscope (along with a custom-designed current amplifier circuit) has been utilized to monitor the voltage drops in the mote’s power supply during computation and wireless transmission. A signal has been used in both software and hardware implementations of the applications to measure the execution time; this signal transits to high when the execution of a specific task starts and, then, toggles back to low, when that process ends. The energy consumption is calculated using the integral of the measured voltage $V_m$ for the measured execution time period $\Delta \tau$. To calculate the reference current $I_{ref}$ the integral is divided with the reference resistance $R_{ref}$ of our amplifier circuit, which is equal to 0.1Ω. The overall energy consumption is calculated, based on formula (1) where the reference voltage $V_{ref}$ is equal to 2.7 V for the Mote and 3.3 V for the CPLD.

$$E = I_{ref}V_{ref}, \text{where } I_{ref} = \frac{\sum V_{m,i} \Delta \tau}{R}$$ (1)

To evaluate the efficiency of our innovative platform, we have implemented three different, yet very widely used, WSN processing tasks which are described in the next subsections.

A. Forward error correction

Typically, WSNs are utilizing multi-hop topologies. This fact, along with the limited computational power and energy resources of the nodes, has led to the adoption of rather simple error control schemes (some form of Automatic Repeat Request (ARQ)). In noisy environments this approach leads to excessive packet retransmissions, which as it has been demonstrated in [15] can consume most of the available energy in a node. At the same time star (i.e. single-hop) topologies are starting to be introduced in multiple WSNs for a number of reasons such as simpler routing, low (or no) protocol overhead, increased reliability and low delay[16].

In such a network topology, Forward Error Correction (FEC) schemes can be efficiently utilized in order to minimize the required retransmissions; end nodes need only to carry out the relatively light task of encoding, while powerful and less energy-constrained base stations can handle the decoding task. These base stations can also use their higher transmission power in order to communicate with the end nodes. As shown in [16], this organization can lead to a more energy-efficient sensor network than a standard multi-hop one.

To demonstrate the efficiency of our approach we have employed a star topology and implemented a UMTS-like turbo encoder [17] in our reconfigurable platform while the decoder is implemented in the base station (a PC with a ZigBee dongle in our case). Comparing to the code described in [17], the following simplifications were made to reduce the implementation complexity and tailor the scheme to the needs of the WSN applications: (a) input length is fixed to 160 bits (which is sufficient in practice according to [15]), (b) a simple relative prime interleaver is used and (c) no puncturing is applied to the data. The proposed Turbo Coder is a Parallel Concatenated Convolutional Code with two 8-state constituent encoders and an internal interleaver. Code rate is 1/3. All the encoding process is handled by the CPLD and only the information packing is performed in the micro-controller.

Table I demonstrates the execution time and the energy consumption when the Turbo Encoder task is implemented in both a standard commercial mote and in our novel platform. The results indicate that for this application only a slight reduction in the overall execution time is triggered by our platform. The overhead is due to the very limited intercommunication
bandwidth between the CPLD and the mote’s micro-controller, as the Turbo Encoder which has been implemented in the CPLD can perform much faster than the software one (the hardware encoder circuitry is clocked almost 7 times faster than the microcontroller - 48MHz vs 7.2MHz - and can produce an encoded bit on every clock cycle). In particular, most of the reported execution time for our novel platform, is spent in the I/O operations. During this time, the microcontroller performs no computations (even the aforementioned data packing is masked in the data reception process). The fact that the microcontroller is mostly in idle state leads to significant energy reductions when compared to the standard software approach. As the results in Table I demonstrate, our novel platform consumes 70% less energy than the standard IRIS one when executing the exact same task.

### B. Routing

Routing in WSNs is challenging due to the inherent characteristics that distinguish these networks from other wireless ones like mobile ad-hoc or cellular networks. The task of finding and maintaining routes in WSNs is non-trivial since energy restrictions and sudden changes in node status (e.g. failure) cause frequent and unpredictable topological changes. In order to minimize the overall energy consumption several routing techniques proposed in the literature employ well-known WSN-specific routing tactics, such as data aggregation and in-network processing, clustering, different node role assignment as well as certain data-centric methods ([1]).

TinyOS, which is probably the most widely used OS in WSNs, supports natively three different routing protocols: Route, MINTRoute and ReliableRoute (XMESH) [18], [19], [20], [21], [22], [23]. The most commonly used routing scheme is XMESH mainly due to its performance when applied to real-world WSNs. In order to form and maintain an XMESH network, the following two parallel processes are involved: Link Estimation and Parent Selection. Those processes calculate certain cost metric functions based on certain multiplications; after profiling them we realized that those multiplications were indeed used very often in the calculation of both metrics and therefore we have implemented the multiplication tasks in the CPLD. Table I illustrates the performance and the energy consumption when both our novel platform and a standard commercial mote are executing the XMESH routing protocol stack. The results derived from real-world experiments where up to 12 nodes formed a mesh network and up to 1000 packets were exchanged among them. The presented results are the arithmetic mean values of the above measurements whereas the variant is insignificant.

As the table demonstrates, the execution time in both our proposed platform and the standalone Crossbow mote are the same and this is due to the significant overhead introduced by the CPLD-mote communication as described previously. More importantly, the energy consumption of the proposed scheme is almost 72% lower than that of the commercial mote.

### C. Security

Security is a critical factor for numerous WSN applications. Motes, however, have limited energy resources and computational capabilities. Thus, to apply certain security schemes, the designer faces the seemingly contradictory challenge of providing complex algorithms for strong authentication and encryption that can be performed reasonably fast and efficiently from an energy consumption standpoint.

Blowfish [24] is a secret-key block cipher with a 64-bit block size. It is based on a Feistel network employing a simple encryption function 16 times and can use variable-length keys. There are also mini versions of Blowfish, such as Blowfish-16, which can operate on 16-bit blocks. Its small block size makes this version of Blowfish perfectly suitable for WSNs. Both the encrypter and decipher processes of the Blowfish algorithm have been implemented in the CPLD to evaluate our platform for WSNs that have high security requirements. Since the hardware implementation of the Blowfish cipher encryption supports only 16-bit blocks, a Cipher Block Chaining (CBC) task is implemented in software to efficiently encrypt/decrypt as many data bytes as needed. This enables us to support variable lengths of plaintext of up to 128 bits. CBC is supported for both encryption and decryption.

To have a better insight on the efficiency of our approach, we have implemented four different versions of the Blowfish scheme which differ only in the block size (16, 32, 64 and 128 bits). Table II illustrates the deriving results. These demonstrate that the time needed to calculate the final result (ciphertext or plaintext depending on the procedure) from our platform is significantly more than that required by the standalone mote executing the same algorithms. Similarly to the Turbo encoding case presented above, the CPLD - mote intercommunication is mainly responsible for this overhead. The impact, however, is much more severe in this application, since the amount of information that has to be exchanged between the mote and the CPLD is much larger.

However, and more importantly, our platform is much more energy efficient when implementing the Blowfish scheme than the Crossbow’s motes. For the encryption procedure, the proposed scheme consumes around 90% less energy than that consumed by a MICAz executing the exact same task while for the decryption procedure the reduction in energy consumption achieved by our novel node platform is more than 80%.

### V. Conclusions

In this paper, we present a novel platform for wireless sensor network nodes which incorporates reconfigurable hardware devices. By introducing a low-power and low-cost reconfigurable

### TABLE I

<table>
<thead>
<tr>
<th></th>
<th>Execution time (ms)</th>
<th>Energy consumption (uJ)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Mote</td>
<td>Mote/CPLD</td>
</tr>
<tr>
<td>Turbo Encoding</td>
<td>7.2</td>
<td>6.6</td>
</tr>
<tr>
<td></td>
<td>(Reduction: 69.8%)</td>
<td></td>
</tr>
<tr>
<td>XMESH Routing</td>
<td>125.0</td>
<td>123.0</td>
</tr>
<tr>
<td></td>
<td>(Reduction: 71.5%)</td>
<td></td>
</tr>
</tbody>
</table>
device, we significantly reduced the energy consumption of a widely used WSN node. Several real-world experiments demonstrate that our platform is between 70% and 93% more energy efficient than the most widely used commercial WSN mote. These energy savings are offered either at the same performance levels (except for a certain application that a slight reduction in performance is observed). The reason behind the slight lack of performance of our system has been identified and it is common in all cases: it is the low bandwidth behind the slight lack of performance of our system has been identified and it is common in all cases: it is the low bandwidth.

**TABLE II**

<table>
<thead>
<tr>
<th>Block size (bits)</th>
<th>Execution time (us)</th>
<th>Energy consumption (uJ)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Mote/CPLD</td>
<td>Mote/Mote</td>
</tr>
<tr>
<td><strong>Encryption</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>75.0</td>
<td>79.8</td>
</tr>
<tr>
<td></td>
<td>106.6</td>
<td>7.2</td>
</tr>
<tr>
<td>32</td>
<td>169.2</td>
<td>546.0</td>
</tr>
<tr>
<td></td>
<td>182.0</td>
<td>14.0</td>
</tr>
<tr>
<td>64</td>
<td>337.0</td>
<td>954.0</td>
</tr>
<tr>
<td></td>
<td>313.6</td>
<td>29.5</td>
</tr>
<tr>
<td>128</td>
<td>955.0</td>
<td>2735.0</td>
</tr>
<tr>
<td></td>
<td>507.4</td>
<td>58.0</td>
</tr>
<tr>
<td><strong>Decryption</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>48.9</td>
<td>136.8</td>
</tr>
<tr>
<td></td>
<td>47.1</td>
<td>8.8</td>
</tr>
<tr>
<td>32</td>
<td>99.4</td>
<td>275.5</td>
</tr>
<tr>
<td></td>
<td>96.7</td>
<td>17.0</td>
</tr>
<tr>
<td>64</td>
<td>198.0</td>
<td>555.0</td>
</tr>
<tr>
<td></td>
<td>189.7</td>
<td>34.3</td>
</tr>
<tr>
<td>128</td>
<td>354.0</td>
<td>984.0</td>
</tr>
<tr>
<td></td>
<td>343.6</td>
<td>56.3</td>
</tr>
</tbody>
</table>

**ACNOWLEDGMENTS**

This work is part of the Secure, Mobile visual sensor networks ArchiTecture (SMART, project number: 100032) research project, funded within the ARTEMIS Joint Technology Initiative as a Joint Undertaking project between the European Commission, the member states and ARTEMIS Industrial Association (ARTEMISIA).

**REFERENCES**


