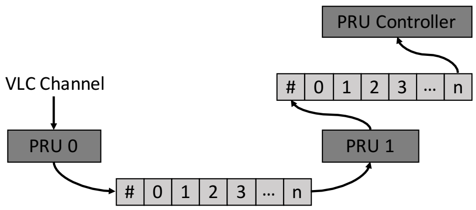


Fig. 4. Memory sharing between the PRUs, and between PRU_1 and the PRU controller ('#' stores the physical address of the latest updated data).

2) Signal reception: While receiving, PRU_0 captures data from ADC and stores it to a shared memory with PRU_1. This memory is circular and continuously updated. The address of the latest measurement is placed at the beginning of the shared memory as illustrated in Fig. 4. Based on this information, PRU_1 reads the data and processes it. Once a frame is totally decoded, PRU_1 sends it to the PRU controller through a shared memory, similar to the one used in the TX. This is also illustrated in Fig. 4.

V. Preliminary Evaluations

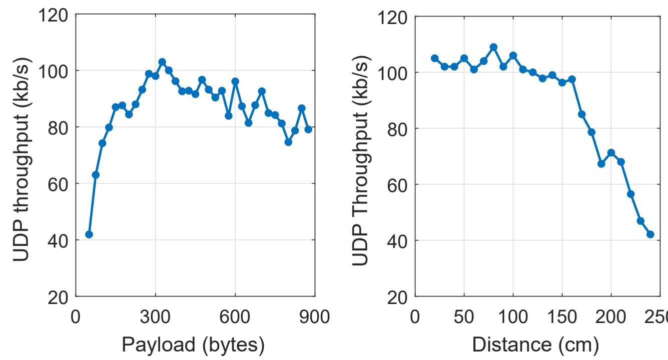
In this section we evaluate the performance of OpenVLC1.2.

Setup. We use two OpenVLC1.2 nodes, one as a transmitter and the other as a receiver. For the optical devices, we use the SENCART 2W LED and the SFH-206 (PD). The sampling frequency at the receiver is 200 MHz. The speed of modulating the LED light at the transmitter is set to the same frequency. Since OpenVLC1.2 provides a new network interface that can be easily accessed by upper layer applications, we use the tool *iperf* to evaluate the UDP performance of OpenVLC1.2.

Throughput vs. payload. We place the receiver at a distance of 20 cm from the transmitter. We vary the UDP payload from 50 to 900 bytes. The evaluation results are shown in Fig. 5(a). We can observe that the UDP throughput first increases then decreases with the increasing of payload. If the payload is small, the throughput is low due to the relative large overhead in the frame headers. When the payload is large, the probability of wrongly decoding a frame also increases, which degrades the achievable system throughput. Nevertheless, we can see that the maximum UDP throughput achieved by OpenVLC1.2 can reach 100 kb/s when the payload is 300 bytes, increasing the throughput over its predecessors by around 8x.

Throughput vs. distance. In this scenario, we fix the payload to 300 bytes. We vary the distance between the transmitter and the receiver to evaluate the performance of OpenVLC1.2 with respect to distance. The results are shown in Fig. 5(b). We can see that the UDP throughput remains around 100 kb/s at distances up to 150 cm. After that, the signal strength becomes weaker and the UDP throughput starts to decrease.

Limitations. The throughput and transmission distance in OpenVLC1.2 can be still largely improved with better front-end design. Nevertheless, the achieved 100 kb/s UDP throughput in OpenVLC1.2 is fast enough for several IoT applications.



(a) UDP throughput vs. payload (b) UDP throughput vs. distance

Fig. 5. Preliminary evaluations of the OpenVLC1.2 platform

VI. Conclusion

In this paper, we presented the design and preliminary performance evaluation of OpenVLC1.2, an open-source platform designed to lower the barriers to VLC research for the Internet of Things. To the best of our knowledge, OpenVLC1.2 is the first open-source platform that achieves a UDP throughput of 100 kb/s using only low-end hardware. Apart from being used for research and teaching as its predecessors, OpenVLC1.2 can enable applications in real world. Going forward, we will continue to improve the performance of OpenVLC to benefit the research community.

Acknowledgment

This work has been funded in part by the Madrid Regional Government through TIGRE5-CM (S2013/ICE-2919) and in part by the "La Caixa international PhD program" fellowship.

References

[1] "pureLiFi," *https://purelifi.com/*, 2018.
[2] J. Zhang, X. Zhang, and G. Wu, "Dancing with light: Predictive in-frame rate selection," in *Proc. IEEE INFOCOM*, 2015, pp. 1–9.
[3] C. B. Liu, B. Sadeghi, and E. W. Knightly, "Enabling vehicular visible light communication (V2LC) networks," in *Proc. VANET*, 2011.
[4] "WARP Project," *http://warpproject.org*, 2018.
[5] "Universal Software Radio Peripheral," *https://www.ettus.com/*, 2018.
[6] N. O. Tippenhauer, D. Giustiniano, and S. Mangold, "Toys communicating with leds: Enabling toy cars interaction," in *IEEE CCNC*, 2012.
[7] T. Li, C. An, Z. Tian, A. T. Campbell, and X. Zhou, "Human sensing using visible light communication," in *ACM MobiCom*, 2015.
[8] C. Zhang, J. Tabor, J. Zhang, and X. Zhang, "Extending mobile interaction through near-field visible light sensing," in *ACM MobiCom*, 2015.
[9] Y. Kuo, P. Pannuto, K. Hsiao, and P. Dutta, "Luxapose: Indoor positioning with mobile phones and visible light," in *ACM MobiCom*, 2014.
[10] C. Zhang and X. Zhang, "Litell: Robust indoor localization using unmodified light fixtures," in *ACM MobiCom*, 2016.
[11] Q. Wang, M. Zuniga, and D. Giustiniano, "Passive communication with ambient light," in *ACM CoNEXT*, 2016.
[12] X. Xu, Y. Shen, J. Yang, C. Xu, G. Shen, G. Chen, and Y. Ni, "PassiveVLC: Enabling Practical Visible Light Backscatter Communication for Battery-free IoT Applications," in *ACM MobiCom*, 2017.
[13] Q. Wang, D. Giustiniano, and D. Puccinelli, "OpenVLC: Software-Defined Visible Light Embedded Networks," in *ACM VLCS*, 2014.
[14] "BeagleBone Black," *http://beagleboard.org/Products/BeagleBone+Black*, 2018.