


# IoT for Real-Time Measurement of High-Throughput Liquid Dispensing in Laboratory Environments

SLAS Technology  
2018, Vol. 23(5) 440–447  
© 2018 Society for Laboratory  
Automation and Screening  
DOI: 10.1177/2472630318769454  
journals.sagepub.com/home/jla  


Justin Shumate<sup>1</sup>, Pierre Baillargeon<sup>1</sup>, Timothy P. Spicer<sup>1</sup>,  
and Louis Scampavia<sup>1</sup>

## Abstract

Critical to maintaining quality control in high-throughput screening is the need for constant monitoring of liquid-dispensing fidelity. Traditional methods involve operator intervention with gravimetric analysis to monitor the gross accuracy of full plate dispenses, visual verification of contents, or dedicated weigh stations on screening platforms that introduce potential bottlenecks and increase the plate-processing cycle time. We present a unique solution using open-source hardware, software, and 3D printing to automate dispenser accuracy determination by providing real-time dispense weight measurements via a network-connected precision balance. This system uses an Arduino microcontroller to connect a precision balance to a local network. By integrating the precision balance as an Internet of Things (IoT) device, it gains the ability to provide real-time gravimetric summaries of dispensing, generate timely alerts when problems are detected, and capture historical dispensing data for future analysis. All collected data can then be accessed via a web interface for reviewing alerts and dispensing information in real time or remotely for timely intervention of dispense errors. The development of this system also leveraged 3D printing to rapidly prototype sensor brackets, mounting solutions, and component enclosures.

## Keywords

IoT, dispenser, HTS, microcontroller, 3D printing

## Introduction

Modern high-throughput screening (HTS) makes use of pressurized-bottle solenoid valve dispensers to rapidly transfer small volumes of liquid into microtiter plates of varying densities.<sup>1</sup> These devices provide accurate liquid-handling capabilities and serve as one of the foundations of HTS. These devices are not infallible, though, and equipment issues can occur. Valves and tips used on these HTS dispensers are considered consumable items intended for periodic replacement when they fail. The primary modes of failure include blocked dispense tips and failed dispense valves, which lead to incorrect volumes being dispensed. Undetected, these errors can lead to significant losses in time or reagents during screening. Therefore, there is a desire for real-time quality control (QC) of these processes as they occur.<sup>2–5</sup>

One QC solution to this challenge is manual operator intervention. A human operator can pause the robotic system temporarily to remove the microtiter plate in question and transport it to an analytical balance. Once at the balance, the operator is able to take weight measurements before and after dispenses to ensure their accuracy. This method has the disadvantage, however, of slowing down throughput and potentially introducing human error. A second approach to this

problem involves robotically integrating inline process-monitoring systems to verify the quality of dispenses.<sup>6,7</sup> This method involves the use of designated weigh stations installed in the HTS robotic environment. These stations allow the robot to weigh plates before and after reagent dispenses to ensure the correct volume of reagent is being added at each step. This means that for each individual dispense step, the plate must be moved to a stand-alone weigh station twice, once prior to dispensing and then once again after dispensing. The downside of this method is that these movements to and from discrete weigh stations take additional time and potentially create bottlenecks in the HTS system.

The automated weighing system presented here, referred to internally as Just Another Monitoring System (JAMS),

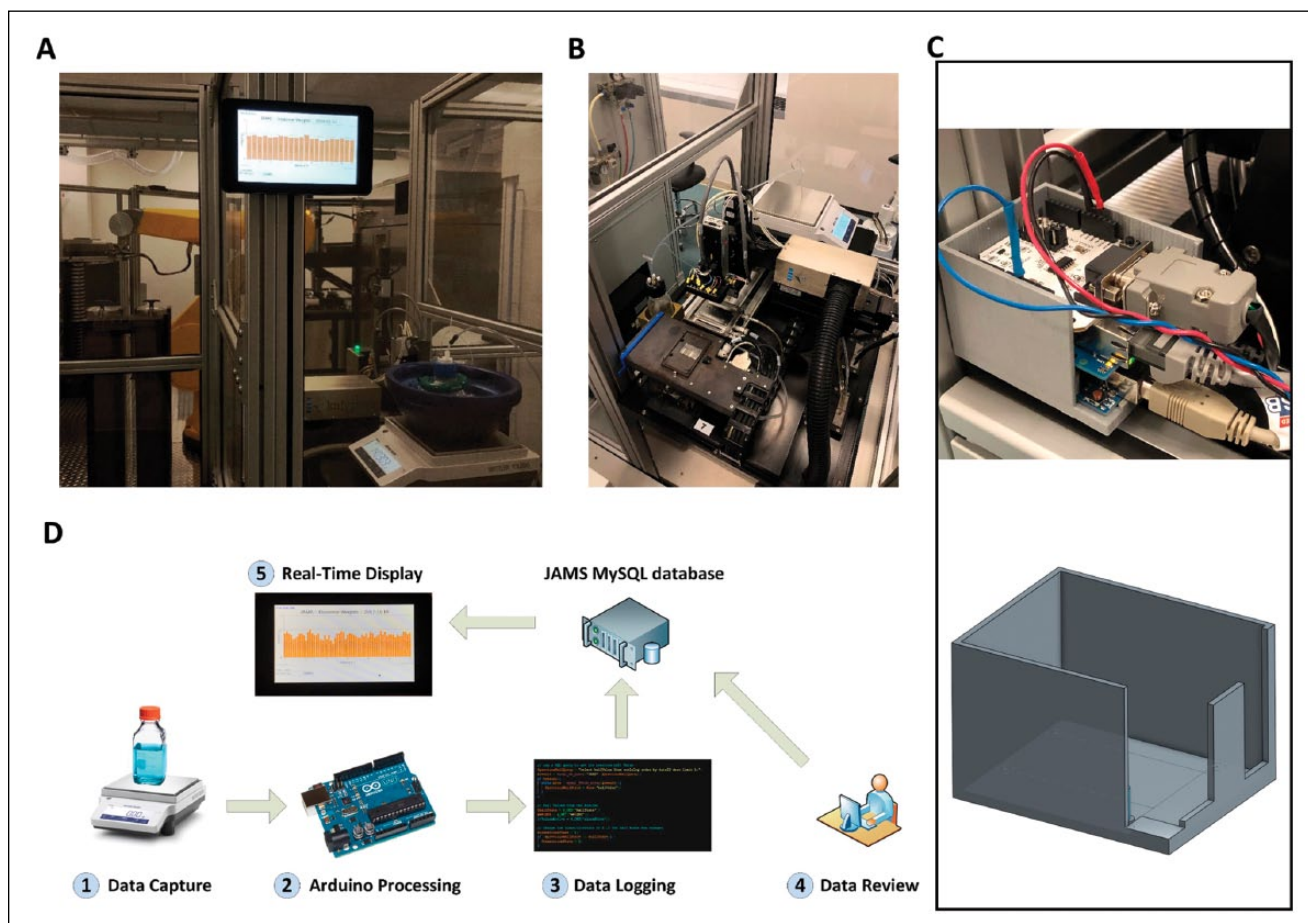
---

<sup>1</sup>The Scripps Research Institute Molecular Screening Center and Department of Molecular Medicine, Scripps Florida, Jupiter, FL, USA

Received January 30, 2018.

### Corresponding Author:

Louis Scampavia, The Scripps Research Institute Molecular Screening Center and Department of Molecular Medicine, Scripps Florida, 130 Scripps Way #1A1, Jupiter, FL 33458, USA.  
Email: scamp@scripps.edu



**Figure 1.** (A–C) View of the Just Another Monitoring System (JAMS) dispenser QC (quality control) platform showcasing the balance, Arduino microcontroller, Raspberry Pi display, photo interrupter sensor, and dispenser plate positioning stage. Also includes a computer-aided design (CAD) model of a 3D-printed case used to store the Arduino microcontroller, Adafruit Ethernet Shield, and LinkSprite RS232 Shield in a stack. (D) Overview of the JAMS process, with data being read from a balance, captured by an Arduino microcontroller, stored into a database, and made available for review via a web interface.

represents an Internet of Things (IoT)-based solution to address these problems. This system leverages open-source software and electronics, alongside 3D printing, to implement a novel automated dispense-weighing system without the requirement of discrete robot-accessible weigh stations. Open science solutions have proven to be an area of interest, with many investigators in the field discovering opportunities to solve problems that previously required expensive proprietary solutions.<sup>8–16</sup>

The JAMS platform, an overview of which is seen in **Figure 1**, accomplishes the goal of dispense QC by integrating the weighing process directly with the reagent bottles, which are mounted on a precision balance. A sensor is mounted to the dispenser platform to detect when dispenses occur, and the data from the balance is continuously output and stored by an Arduino<sup>17</sup> into a database. After a dispense operation is finished, the database is queried, and the weight before and after is compared to gravimetrically obtain the

amount of liquid dispensed. The key advantage of this system is it does not disrupt the HTS process. No additional automated steps are needed to measure the dispense volume. Because the weight is processed immediately after the dispense finishes, any errors in the measurement can be detected in real time, and error notifications can be sent to relevant personnel. Finally, the system offers an easy method of data review. A web-based interface allows easy access to interact with and monitor the system over any local computer or mobile device that is connected to the Scripps Network. With all the balance data stored in a database, the data can be viewed at a later time to review dispenser function throughout the course of an assay.

## Materials and Methods

The JAMS QC system was built using off-the-shelf open-source hardware listed in **Table 1**. Detailed information on

**Table 1.** Parts list for JAMS.

Vendor	Part	Catalog No.	Cost
Adafruit	Adafruit Ethernet Shield R3	201	\$45.00
Amazon	Raspberry Pi 7-inch touchscreen case	FBA_102035	\$18.81
Amazon	CanaKit 5V 2.5A power supply	DCAR-RSP-2A5	\$9.99
Fisher Scientific	Precision Balance—Mettler Toledo	ME3002E	\$1241.25
Sparkfun	Arduino Uno—WR3	DEV-11021	\$24.95
Sparkfun	6V 2A power supply	TOL-14158	\$4.95
Sparkfun	LinkSprite RS232 Arduino Shield, version 2	DEV-13029	\$11.95
Sparkfun	Photo interrupter	SEN-09299	\$1.95
Sparkfun	Sparkfun photo interrupter breakout board	BOB-09322	\$1.50
Sparkfun	Raspberry Pi 3	DEV-13825	\$39.95
Sparkfun	16 GB micro SD card	COM-13833	\$19.95
Sparkfun	Raspberry Pi 7" LCD touchscreen	LCD-13733	\$64.95
Sparkfun	USB micro-B cable	CAB-10215	\$4.95
<b>TOTAL</b>			<b>\$1490.15</b>

how each of these technologies was integrated with the GNF Washer/Dispensers used on the Scripps Florida HTS platform can be found in the subsequent sections below.

## Hardware

The balance used in our implementation is a Mettler Toledo ME3002E, which has been set up to output its weight measurement through the onboard RS232 serial connector. Data from the balance is read by an Arduino Uno microcontroller every second via a LinkSprite RS232 Arduino Shield, which is connected to the RS232 port of the balance. The data stream from the balance contains the numerical value of the weight measured, the measurement units, the sign of the measurement, and the stability of the measurement.

A method was needed to allow the software to determine when a dispense operation starts and finishes. On the GNF dispenser, the platform that holds the microtiter plate during dispensing travels back and forth along a *y*-axis. By placing an infrared (IR) photo interrupter sensor in a 3D-printed mount at one end of the *y*-axis and attaching a 3D-printed flag to the dispenser platform (**Fig. 2**), we are able to tell when the dispenser is at an idle position versus when the plate stage has moved and is actively dispensing. The 3D-printed flag protrudes down in a manner that blocks the sensor when the dispenser is in its idle state and leaves the sensor unobstructed during dispensing. With this information, the software knows which time points from the balance dataset correspond to the start and end of a dispense operation, enabling the calculation of a gravimetric comparison between the two.

Customization of 3D-printed parts was accomplished using the Onshape CAD modeling tools.<sup>18</sup> The Arduino Uno, Ethernet Shield, and RS232 Shield are all stacked together and housed in a 3D-printed enclosure inspired by a

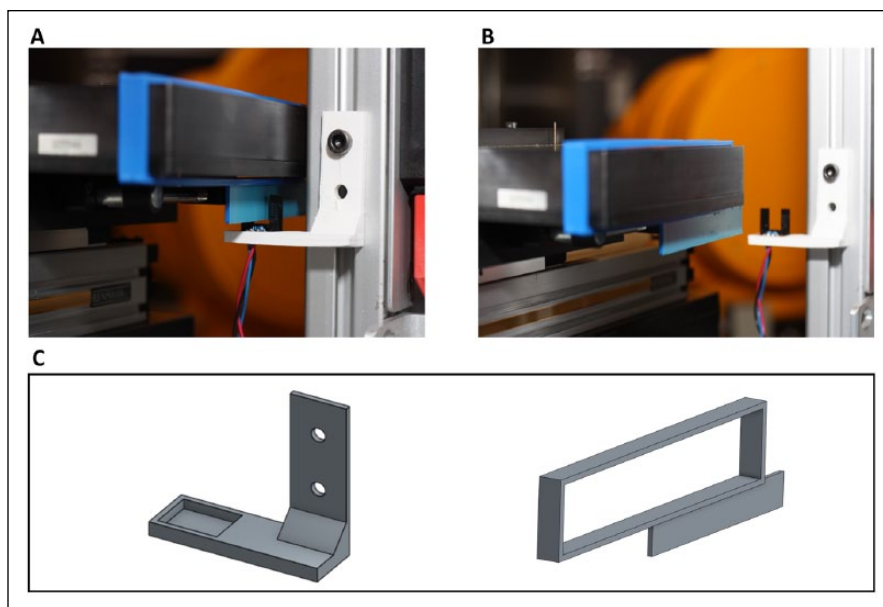
user design found on Thingiverse.<sup>19</sup> All 3D-printed components of the JAMS platform can be seen in **Figures 1C and 2C**.

After collecting the necessary information from the balance and the IR sensor, the Arduino transmits the data to a PHP<sup>20</sup> webpage via an Adafruit Ethernet Shield, which connects the balance to the Scripps Florida intranet. The PHP webpage then takes the data, deposits it into a MySQL<sup>21</sup> database, and determines if any alerts should be generated due to dispense errors. Alerts are currently sent via email and contain specific information regarding the target dispense weight versus the measured weight. This data allows for a clear picture of when dispenses have started, when they are in progress, and when they have finished (**Fig. 3**).

To further improve the real-time monitoring aspect of the dispenser QC system, a Raspberry Pi<sup>22</sup> and 7-inch touchscreen were mounted directly above the dispenser (**Fig. 1A**). The Raspberry Pi automatically loads the plate-by-plate dispense QC summary webpage on system initialization. This allows for a visualization of the data coming off the balance to be viewed directly at the dispenser itself, rather than at a separate computer. Because the Raspberry Pi is dedicated to this specific function, it always has the relevant information open and updated for the current dispense task on hand.

## Software

Three PHP programs handle the data storage, retrieval, and visualization tasks of the JAMS system. The first of these three, Data Logger, is responsible for accepting data from the Arduino and storing it into the MySQL database. Data Logger also performs limit checks to see if dispenses are within set parameters and generates appropriate user notifications if they are not. The other two programs, Data



**Figure 2.** (A) 3D-printed flag blocking the photo interrupter sensor, indicating that the dispenser is currently in its idle position. (B) Active dispensing position with flag no longer blocking the path of the photo interrupter sensor. (C) Computer-aided design (CAD) models of 3D-printed components used for photo interrupter sensor mounting and the dispenser platform flag.

Display and Data Review, are used by end users to review data. All three programs interact with a MySQL database that acts as a repository for dispense QC information.

Data Display and Data Review are two visualization JAMS webpages that can be used to analyze the QC data. These different visualizations reflect different goals, with Data Display providing a direct visualization of the raw data from the precision balance, and Data Review providing a per-plate weight change visualization. Both webpages use a JavaScript graphing system called CanvasJS<sup>23</sup> to generate the dynamic charts used in the JAMS visualizations.

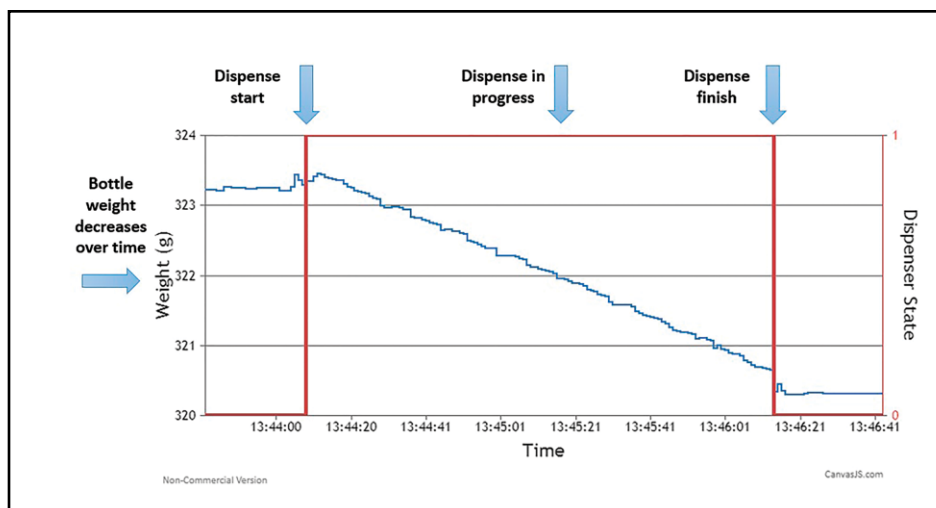
The Data Display page, seen at the bottom of **Figure 4**, is created as a visualization of the real-time raw data from the balance and the IR photo interrupter sensor. The data is displayed as a step line graph with dual axes for the weight and the sensor

value. The weight axis is automatically scaled to the relevant data. The sensor data is scaled between 0 and 1 because these are the only two discrete states that the sensor can take.

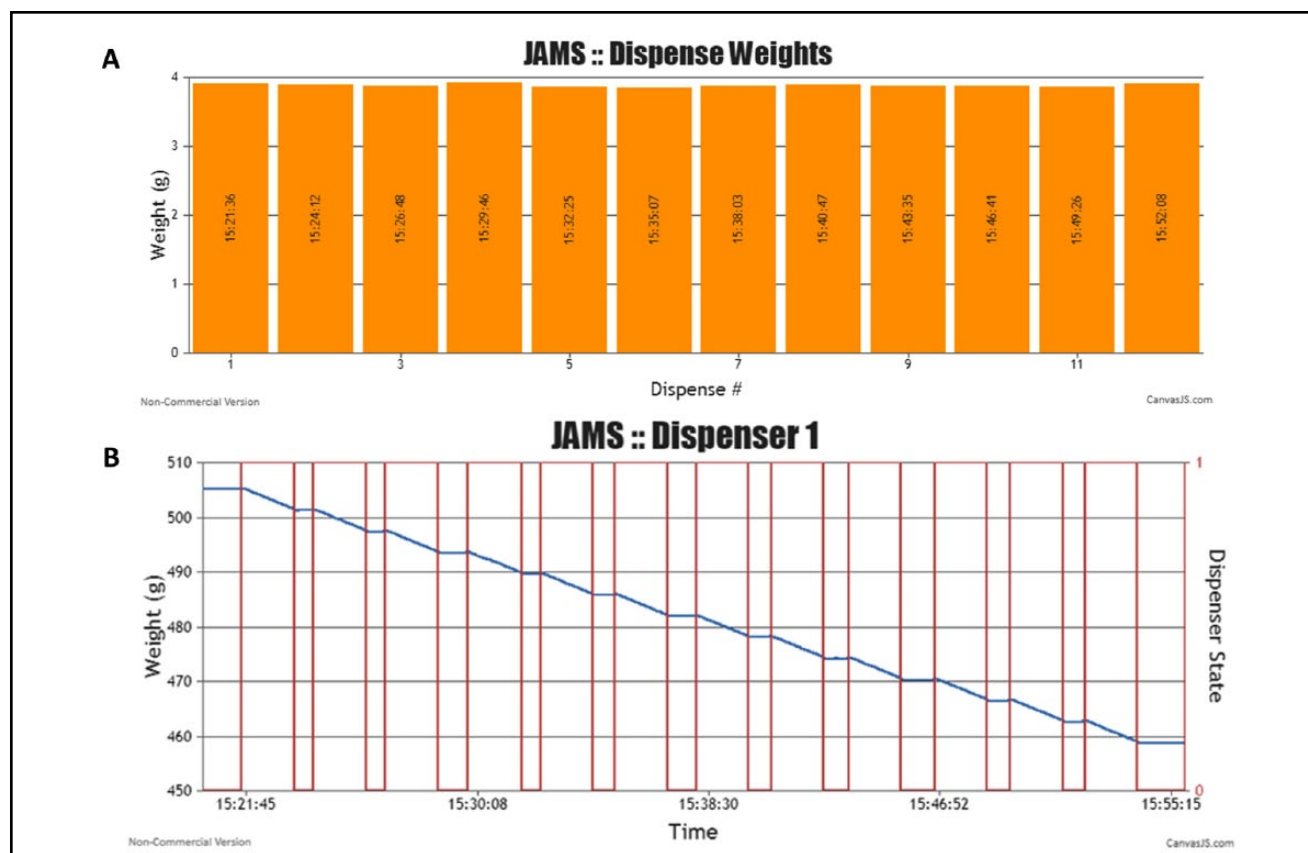
The Data Review page, seen at the top of **Figure 4**, is a representation of the QC data in relative plate-by-plate format. Dispenses shown on the Data Review page record both the number of dispenses and their mass transfers. The data is represented as a bar graph, with each bar giving the weight of that dispense. Within each bar, the start time of the dispense is displayed.

## Results

To demonstrate the capability of JAMS to provide real-time error detection and reporting of dispense issues, a series of



**Figure 3.** Graph of decreasing weight readout from the balance and sensor state throughout the course of a dispense, corresponding to volume of reagent dispensed into plate.



**Figure 4.** (A) The data review webpage visualization, showing the results of 12 dispenses during the pilot test on a plate-by-plate basis. (B) The same high-throughput screening (HTS) pilot data in a data display visualization, demonstrating the raw values captured from the balance decreasing throughout time and the sensor state changing as dispenses are started and stopped.

experiments was performed. These experiments included dispense operations as part of a standard HTS campaign and a separate simulation of a dispense failure.

The first method of verification for JAMS was a series of pilot tests emulating standard HTS campaigns. In this configuration, the Scripps HTS platform was run as normal without any modifications, apart from the addition of the JAMS platform installed alongside the GNF dispenser. Sample data captured from this phase of testing can be seen in **Figure 4**. At the top of this figure, dispense data from a series of 12 plates is visualized in a relative plate-by-plate dispense weight format. At the bottom of this figure, the raw data from the balance and IR sensor is seen. This dataset represents a series of sequential plate reagent dispenses occurring without any failure. Furthermore, the dataset shows a small plate-to-plate weight coefficient of variation (CV) of less than 5%, which falls within acceptable HTS dispense limits.

During the pilot run, results from the JAMS platform were also compared against a traditional method of QC in which the operator periodically suspended the HTS platform to retrieve plates before and after dispense for

weighing on an offline analytical balance. Results from this comparison for a subset of plates are seen in **Table 2**. The difference between the weights measured by the JAMS platform and the manual operator weighing varied by 1.0% on average, with the largest difference between the two measurements being 2.6%. Data obtained from each balance was also consistent: the %CV of the measurements from the operator weighing was 0.47%, and the automated JAMS weighing was 0.52%. This data shows that the JAMS platform is accurately and consistently measuring the dispense weight as compared to the traditional operator QC.

The next round of testing involved a series of dispense operations followed by a valve failure. The valve failure was simulated by disconnecting the power supply from the valve electronics connector. In **Figure 5**, the data visualization shows two nominal dispenses, seen as linearly decreasing weights between the dispense start and dispense stop signals. Following these first two dispenses, the simulated failure is observed when the measured weight of the plate decreases in the plate-by-plate visualization and the raw weight from the balance in the raw data visualization no longer decreases at the same rate it had in the previous two



**Table 2.** Comparison of Results: Operator Measured versus Automated Balance Measured.

Plate No.	Manual Balance Weight (g)	Automated Balance Weight (g)	Absolute Difference (g)	Relative Difference (%)
1	3.87	3.90	0.03	0.8
2	3.85	3.89	0.04	1.0
3	3.84	3.88	0.04	1.0
4	3.86	3.92	0.06	1.5
5	3.85	3.86	0.01	0.3
6	3.83	3.85	0.02	0.5
7	3.84	3.88	0.04	1.0
8	3.85	3.89	0.04	1.0
9	3.83	3.87	0.04	1.0
10	3.83	3.87	0.04	1.0
11	3.85	3.86	0.01	0.3
12	3.80	3.90	0.10	2.6

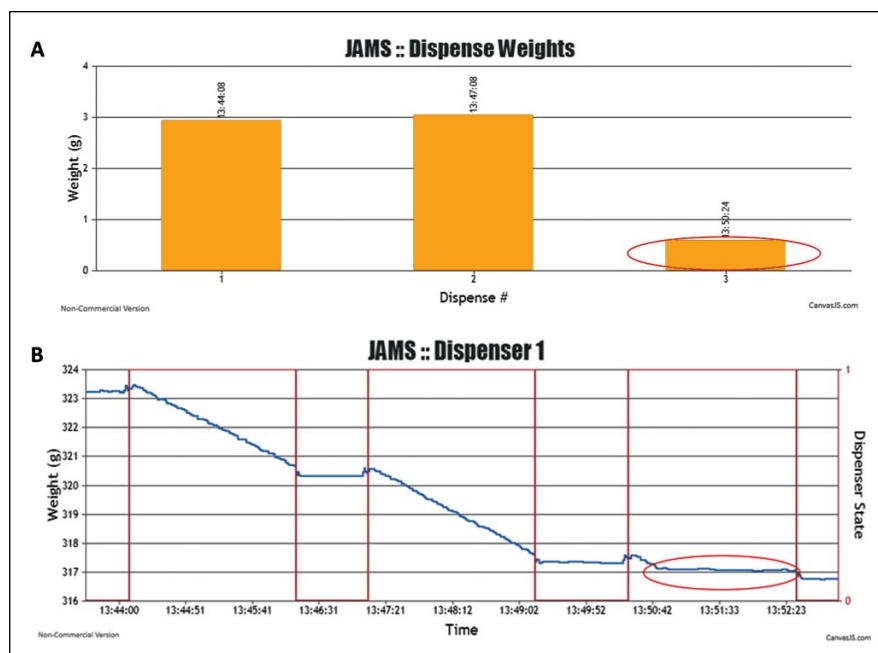
dispenses. This is denoted by the circled data in **Figure 5**. The simulated dispense error triggered an email alert that immediately notified relevant personnel that a dispense error had occurred.

## Discussion

The JAMS platform represents a robust, automated, and inexpensive dispense QC monitoring system for HTS dispensing. This was accomplished by combining low-cost microcontrollers (including an Arduino and Raspberry Pi), a web-based PHP–MySQL interface, an inexpensive IR photo interrupter sensor, 3D-printed parts, and a precision balance into a paradigm-changing IoT device. The total cost

of the platform, including the precision balance, was less than \$1500.

Pilot testing has shown that the JAMS platform provides automated dispenser QC without any interference to the throughput of the robotic platform. In fact, it operates entirely outside of any of the platform's software or hardware utilities, which makes it easily adoptable to other platforms. Data from pilot testing also confirmed that the system was functioning as intended and successfully characterized dispenser performance throughout time. After the completion of an HTS campaign, any discrepancies seen in the experimental results can be cross-referenced with the historical records of the dispensing for that assay. Access to this type of historical information may reveal trends about dispense inconsistencies



**Figure 5.** (A) Pilot test showing an error in a dispense operation in which the third dispense has encountered a simulated valve blockage and no longer outputs the correct volume. (B) Data display visualization of the balance's output, indicating a dispense error.

that can be used to address dispenser performance. Prior to the implementation of JAMS, historical data for dispensing operations at Scripps Florida was available for only a small percentage of HTS plates that had been manually retrieved and weighed by an operator using an offline balance. In addition, the real-time alerts generated by the JAMS platform allow HTS operators to address errors as they occur, potentially saving precious reagents, time, and other consumables. The end benefit of these factors is that HTS operations are more efficient and robust.

There are a variety of avenues for potential future development of the JAMS platform. These include increasing the number of precision balances connected to the IoT platform to support a greater number of dispensers in parallel. The current system uses one balance with one dispenser for monitoring liquid dispensing. Additional balances could allow for monitoring of multiple dispensers simultaneously, or multiple reagents dispensing from the same dispenser simultaneously. The existing informatics infrastructure can support this upgrade with minimal changes.

The sophistication of the JAMS informatics processing could also be increased by adding algorithms to evaluate the dispense rate from start to finish. Currently, the algorithm relies on endpoint measurements from the start and end of a dispense to determine the whole plate dispense weight. In theory, the data from a properly functioning dispense operation will have a constant, decreasing slope when tracking the weight. A slope that changes during the course of a single dispense could be automatically detected, identifying nonuniform dispense patterns.

A tighter integration between the JAMS platform and the HTS software, including the scheduler and dispenser application, would allow for easier determination of expected dispense weights without separate operator-defined parameters. This could be accomplished via a software application programming interface (API) provided by the HTS robotics vendor, or the system could be adapted to monitor HTS robotics network traffic to determine what protocol is being run. Moreover, either the protocols themselves could then be read, or a list of protocols could be stored in a database table that could be queried to determine if the resulting dispense volume is acceptable relative to the targeted dispense volume. Another potential advantage of tighter integration would be the ability to make automatic corrections in real time to adjust for any errors in the dispense weight. This system would incorporate feedback control, and it would alter the parameters of the dispense method on the fly to bring the dispense weight of future dispense operations closer to the target weight.

Introducing network connectivity to previously offline devices transforms them into IoT-enabled systems, improving the overall functionality of these devices and enabling users to have greater control over their operation. Logging the results from such systems allows for both real-time

analysis and postassay review of the data. Trends in dispense performance can be tracked throughout extended periods of time. As a practical matter, the implementation of the JAMS platform demonstrates this by adding fully automated data logging and alert generation to a system that previously relied on intermittent and manual processing performed by the robotic operator. The end effect of this systematic change is an overall increase in efficiency of laboratory function.

### Declaration of Conflicting Interests

The authors declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

### Funding

The authors received no financial support for the research, authorship, and/or publication of this article.

### References

1. Cox, B.; Denver, J. C.; Binnie, A.; et al. Application of High-Throughput Screening Techniques to Drug Discovery. *Prog. Med. Chem.* **2000**, *37*, 83–133.
2. Curtis, R.; Rodrigues, G.; Artel. Comparing Calibration Technologies for Liquid-Handling Quality Assurance. <http://www.artel-usa.com/resource-library/comparing-technologies-for-pipette-calibration/>.
3. Baillargeon, P.; Scampavia, L.; Einsteder, R.; et al. Monitoring of HTS Compound Library Quality via a High-Resolution Image Acquisition and Processing Instrument. *J. Lab. Autom.* **2011**, *16*, 197–203.
4. Genetic Engineering & Biotechnology News. Machine Vision Improves HTS Quality Control. <https://www.geneng-news.com/gen-articles/machine-vision-improves-hts-quality-control/2412>.
5. Cassaday, J.; Shah, T.; Berry, K.; et al. In “Real Time” Process Control of HTS Assays Using a Vision Inspection System. Proceedings of the Laboratory Automation Conference 2008, Palm Springs, CA, 2008.
6. HighRes Biosolutions. <http://highresbio.com/components/plateweigh.php>.
7. GNF Systems. <http://gnfsystems.com/gallery/>, <http://gnfsystems.com/wp-content/uploads/2013/10/Plate-Balance.jpg>.
8. Il, J. M. G.; Hart, A. S. Opening New Frontiers in the Development of Life Sciences Technology with Collaborative 3D Printing Technology. *J. Lab. Autom.* **2016**, *21* (4), 487–488.
9. Coakley, M.; Hurt, D. E. 3D Printing in the Laboratory: Maximize Time and Funds with Customized and Open-Source Labware. *J. Lab. Autom.* **2016**, *21*, 489–495.
10. Woelfle, M.; Ollairo, P.; Todd, M. H. Open Science is a Research Accelerator. *Nature Chem.* **2011**, *3*, 745–748.
11. Chiu, S.-H.; Urban, P. L. Robotics-Assisted Mass Spectrometry Assay Platform Enabled by Open-Source Electronics. *Biosens. Bioelectron.* **2015**, *64*, 260–268.
12. Baden, T.; Chagas, A. M.; Gage, G.; et al. Open Labware: 3-D Printing Your Own Lab Equipment. *PLoS Biol.* **2015**, *13* (3), e1002086. <https://doi.org/10.1371/journal.pbio.1002086>.

13. Gross, B. C.; Erkal, J. L.; Lockwood, S. Y.; et al. Evaluation of 3D Printing and Its Potential Impact on Biotechnology and the Chemical Sciences. *Anal. Chem.* **2014**, *86*, 3240–3253.
14. Pearce, J. M. Quantifying the Value of Open Source Hardware Development. *Modern Econ.* **2015**, *6*, 1–11.
15. Pearce, J. M. *Open-Source Lab: How to Build Your Own Hardware and Reduce Research Costs*. Amsterdam: Elsevier; 2013; p 240.
16. Pearce, J. M. Building Research Equipment with Free, Open-Source Hardware. *Science*. **2012**, *337* (6100), 1303–1304.
17. Arduino. <https://www.arduino.cc>.
18. Onshape. <https://www.onshape.com>.
19. todbot. Arduino+Ethernet Case. **2017**. <https://www.thingiverse.com/thing:21939>.
20. PHP. Hypertext Processor. <http://www.php.net>.
21. MySQL. <https://www.mysql.com>.
22. Raspberry Pi. <https://www.raspberrypi.org>.
23. CanvasJS. HTML5 JavaScript Charts. <https://canvasjs.com>.