Making Embedded NIMEs with Satellite CCRMA
*Featuring the **Raspberry Pi***

ABSTRACT
This workshop focuses on teaching participants how to make embedded musical instruments and embedded installations using Satellite CCRMA. By the close of the six-hour workshop, each participant will complete a new self-contained instrument or installation using a take-home kit, which is about twice the size of a deck of cards. Beginning and intermediate participants will benefit primarily from being led through a series of basic exercises in using the kit, while advanced participants may be most interested in discussing how to extend the kits.

Satellite CCRMA is currently based on the powerful Raspberry Pi embedded Linux board, which executes floating-point instructions natively at 700MHz. Participants will be shown how to run Pure Data (pd) on the board, but participants are welcome to explore other software available on the Satellite CCRMA memory image. Additional topics include Arduino, Firmata, pico projectors, open-source hardware, ChucK, SuperCollider, and more.

Keywords
Embedded musical instruments, Satellite CCRMA, embedded Linux, Pure Data (pd)

1. INTRODUCTION
Many prior NIME projects consist of sensor circuits interconnected with a more powerful computer using an Arduino or other microcontroller. This strategy combines the flexibility of reconfigurable hardware with the computational power provided by a laptop or desktop computer [1]. However, projects built using this kind of platform have the tendency to "die" with time. Some factors leading to the death of projects include forced software updates, changes in future iterations of hardware interfaces and/or cables (e.g. RS-232 serial to USB), etc. These factors are of course avoidable but may require maintenance resources that might not be readily available in the future when needed.

The aim of the Satellite CCRMA project is to enable makers to use a convenient, compact, and *embedded* multi-purpose computer in place of a laptop or desktop computer. The Satellite CCRMA project has been made feasible by the recent availability of relatively inexpensive embedded Linux computers, such as the Beagle Board xM and Raspberry Pi.

Embedded projects offer several advantages [1]. For example, they are self-contained, which makes them more convenient to use. All required cables can be left connected, so they can be demonstrated at a moment's notice simply by taking them off of the shelf and powering them up. In a limited sense, the embedded projects could be considered to be "living" since they are self-sufficient when supplied with power [2]. If appropriate, projects can be kept disconnected from the Internet, so their software will never require updating in order to continue functioning. We believe that this feature is particularly important for instrument makers, who may require years of practice in order to become virtuosos on the new instruments that they design.

The Satellite CCRMA kit can be similarly employed for making installations. The kit can be left in installation spaces without fear of theft (as would be the case if leaving a laptop or smartphone instead). Also, because the kit components are relatively inexpensive, significant numbers of kits can be connected together in a network installation without the cost becoming prohibitive.

Finally, due to the low-cost of creating completed projects, it becomes more feasible for designers to create *libraries* of "living prototypes." In other words, designers can keep older prototypes operational. This can be particularly helpful in the process of iterative design or evaluation by external people.

2. SATELLITE CCRMA
In summary, the Satellite CCRMA project aims to help prototypes live longer by providing a complete prototyping platform in a single, small, stand-alone embedded form factor. Satellite CCRMA is named after Planet CCRMA, an earlier distribution of Linux configured to make it easier to obtain and work with high quality audio in Linux. Similarly, Satellite CCRMA is based on a combination of open-source software and hardware to promote interoperability and future compatibility of prototype components.

This workshop aims to enable the NIME community to use Satellite CCRMA by providing the first on-site Satellite CCRMA workshop at NIME, which will encourage a sharing of ideas, rapid prototyping components, and tools.

3. PLATFORM
An ARM processor running embedded Linux powers the current iteration of the Satellite CCRMA platform. This workshop will employ the Raspberry Pi Model B board, which executes floating-point instructions natively at 700MHz to synthesize sound. The embedded Linux board comes with a Satellite CCRMA memory card, which contains many pre-installed software packages as well as the GNU C/C++ compilers for designers who may wish to compile their own software packages.

During the led part of the workshop, participants will log into their embedded Linux boards via Ethernet and learn to program sound synthesizers in the graphical programming environment Pure Data (see Figure 1 and http://puredata.info). Then, during the following hands-on prototyping session, participants will implement a new musical instrument or installation using their preferred Linux software.

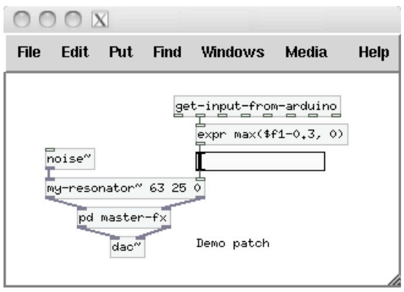

Figure 1. Demonstration patch for sound synthesis in Pure Data (pd)

To facilitate integration with custom sensors, participants are also provided with a breadboard and USB-connected Arduino microcontroller, which sit on top of the embedded Linux board (see Figure 2). Participants are encouraged to bring their own custom sensors for the exploration period, but each participant will also receive a single analog sensor, which can be exchanged and/or shared as desired. By the end of the studio, each participant should complete a simple prototype, such as the prototype, which is shown in Figure 2 and produces sound in headphones when the user squeezes a force-sensitive resistor.

By the end of the workshop, participants will have the knowledge and experience needed to make their prototypes more refined.

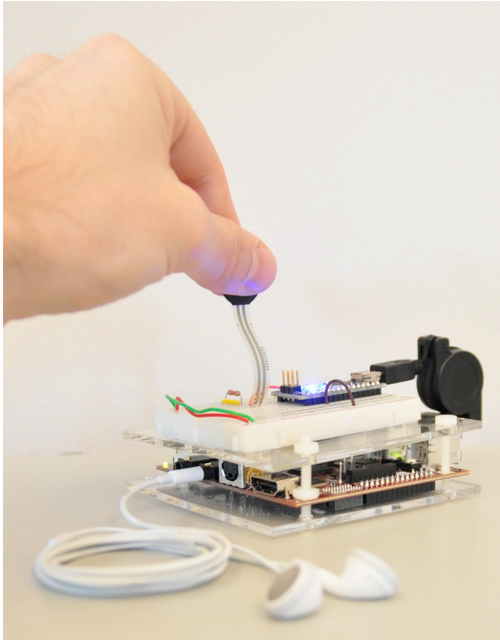

Figure 1. Example prototype based on Satellite CCRMA that intermediate-level participants can expect to complete by the end of the workhop (in this case, the user squeezes a force-sensitive resistor to create sound in earbud headphones)

4. TECHNICAL LEVEL

The technical level of the workshop is intermediate. Participants should be ready to take an interest in learning some aspects of sound synthesis using Pure Data (pd), and prior experience with sensors and the Arduino platform would be beneficial. Each participant should feel comfortable with learning three simple commands in Linux.

5. WORKSHOP TOPICS AND EXPECTED OUTCOMES

Each participant will receive instruction necessary for constructing a new musical instrument using Satellite CCRMA. Relying on videos of prior projects, the workshop presenters will provide advice on what kind of prototypes may be the most successful.

Technical material will be taught by example to address the various levels of technical experience. The practical focus will lie in prototyping sonic interactions using the Pure Data (pd) software. However, significant lecture time will also be devoted to obtaining sensor input via the Arduino (www.arduino.cc) microcontroller and controlling graphical output.

Finally, the following advanced topics will be briefly explored, with an emphasis on providing workshop participants with knowledge about how to get started and where to find more detailed information:

• extending Satellite CCRMA to networked configurations of multiple units,
• integrating audio power amplifiers into prototypes,
• tips on incorporating pico projectors into projects,
• powering prototypes using batteries,
• employing multichannel audio,
• using the platform to compile open-source software packages,
• testing prototypes for robustness and reliability,
• running Satellite CCRMA on other embedded Linux boards,
• and preserving the quality of the flash memory.

6. PROPOSED SCHEDULE

Our Studio will last six hours total and go through three phases: introductions (including assembling the kits), lecture presentation, and hands-on prototyping.

Hour 1 – Introductions: Participants explain their backgrounds and what they would like to get out of the workshop. After showing some videos of prior projects, we will spend 15 minutes constructing the Satellite CCRMA kits out of the kit parts. Finally, we will show participants how to log into the kits from their laptops.

Hours 2 and 3 – During the next two hours, we will introduce participants to sound synthesis in Pure Data (pd). Participants are encouraged to follow along, but some participants with advanced Linux experience may choose to work independently instead. We will also demonstrate how to use Firmata (www.firmata.org) to get sensor data from the attached Arduino microcontroller into pd.[1] Finally, we will introduce advanced topics including the following:

• extending Satellite CCRMA to networked configurations of multiple units,
• employing multichannel audio,
• using the platform to compile open-source software packages,
• powering projects with batteries and audio power amplifiers, and
• running Satellite CCRMA on other embedded Linux boards.

Hours 4 to 6 – During the final three hours, participants will implement their own new musical instrument. Some participants will be most interested in using pd with Arduino and a demo sensor to implement a simple, embedded prototype. Other participants may be interested in compiling additional favorite open-source software packages on the platform. We will assist participants in their endeavors, and participants will also assist each other as appropriate. Finally, following some closing remarks about robustness testing, reliability, and preserving the quality of the flash memory, participants will demonstrate their instruments to each other, we will clean up, and the participants can take their instruments home.

If the conference organizers agree, hours 4 to 6 can be made optional for participants would like to attend a second workshop in the afternoon.

7. COST

Each participant who wishes to take his or her instrument kit home should be prepared to pay the workshop organizers $105 USD in cash (or 80€ in cash or 120,000 KRW in cash). This fee only reimburses for the cost of the parts. Depending on what happens at the Korean customs, the final price could be lower. The workshop owners hope to avoid payment of customs by pre-ordering the most expensive parts and spreading out the parts among the workshop organizers.

We have already initiated discussion with Woon Seung Yeo regarding the possibility of purchasing the Raspberry Pi Model B (512MB) boards directly within South Korea so that each of the workshop organizers can carry less than the customs import limit with entering South Korea.

The total number of kits will be limited to 15. This means that only more than 15 participants should be allowed to enroll if they are willing to share kits.

[1] `StandardFirmata` will be preloaded onto the Arduinos to help simplify the process for any beginners in the workshop.

8. CONCLUSION

Participants will learn about the affordances of Satellite CCRMA and how to harness it for building durable, living prototypes. The practical focus of the workshop is emphasized by requesting that each participant implement a simple new musical instrument or mock-up installation. Unexpected results will arise, particularly when participants demonstrate their prototypes to each other. Following the workshop, participants will consider themselves members of the larger Satellite CCRMA community, where they can go for additional help in the future.

9. REFERENCES

[1] Berdahl, E. and W. Ju. Satellite CCRMA: A Musical Interaction and Sound Synthesis Platform, *Proc. NIME,* 2011, pp. 173-178.

[2] Berdahl, E. and Q. Llimona. Tangible Embedded Linux, *Proc. TEI* (in press), 2013.

[3] Berdahl, E. and C. Chafe. Autonomous New Media Artefacts (AutoNMA). *Proc. NIME*, 2011, pp. 322-323.