Flipping a course on Computer Architecture

Hussein Suleman

Department of Computer Science
University of Cape Town
Private Bag X3, Rondebosch
South Africa
hussein@cs.uct.ac.za

Abstract. This paper reports on an experiment with a flipped classroom for a Computer Architecture course. In a flipped classroom, students access content out of the classroom and then engage in a discussion in-class, rather than the other way around. This seemed like an ideal strategy for a course that can easily focus on the minutiae of architectural details and computer history. The results showed that students liked the interactive and practical aspects of the course but were particularly negative about pre-lecture readings. These results suggest that students need to learn how to learn in different ways, and move away from the exclusive strategy of in-classroom, content-centric lectures.

Keywords: flipped classroom, computer architecture, low resource environments

1 Introduction

In traditional lectures, a lecturer will stand before a class and recite a paper that contains all that is considered important for the class to know, maybe resorting to use of the chalkboard [1]. As technology evolved, this model changed to use presentation cues in Microsoft Powerpoint slides visible to all rather than in notes used only by the lecturer. However, the predominant mode of verbally exposing information to the students remained. This model has been criticized because of its assumption that dissemination of content is the primary purpose of a classroom. It has been argued that, instead, such content is better learnt on an individual basis and the classroom is better utilized as a space for discussion and active engagement with the content [2]. This could lead to higher levels of understanding than if classrooms are used purely for presentation of the content to students.

These arguments are part of the the motivation for flipped or inverted classrooms. A flipped classroom is one where students acquire content knowledge outside the classroom, and use the classroom as a space for discussion instead of lectures[2]. This has been used extensively in teaching and learning in various disciplines, including Computer Science. The traditional format of a flipped classroom includes out-of-classroom video instruction and in-classroom discussions. Students are expected to watch the videos before coming to class to gain the content knowledge and be prepared to engage in discussion about the topics of the videos.

The flipped classroom approach is arguably a good technique because of the following reasons:

- Each student is able to learn at his or her own pace.
- Students are able to learn without explicitly being taught.
- Lectures are more engaging when there is discussion rather than exposition of content.
- Higher levels of understanding are possible if there is greater discussion and interrogation of the content.

This paper reports on an experiment to use the flipped classroom technique in a Computer Architecture course. This course is typically dense in factual content, often leading to traditional forms of lectures. There is little problem-solving and large amounts of technical and historical information. While there is opportunity for debate and discussion, this requires an in-depth knowledge of the content first. Thus, it appeared that this course would be ideal for the flipped classroom approach and this experiment was conceived.

The rest of the paper presents related work, then the design of the course and its various elements, followed finally by an evaluation of the experiment and conclusions.

2 Literature Review

The core idea of an inverted or flipped classroom is presented by Lage, et al. [2], where it is defined as the inversion of activities conducted within the classroom and outside the classroom. Specifically, the format of multimedia or video lectures is presented as a vehicle for the content, with the aim of providing learners with flexibility in their approaches to learning. This simple inversion is often accompanied by an expansion of the range of activities, to supplement the inversion model [3].

In Computer Science, various experiments have been conducted with flipping of courses on different topics, with many recent studies on introductory programming courses.

Campbell, et al. [4] compared a flipped classroom approach to a first year programming course against a traditional version of the same course and demonstrated improvements in the experience of students. They did not mandate compulsory attendance but suggested using in-class quizzes in future years. Latulipe, et al. [5] extended the flipped programming course model for a lab-based course, by adding elements of lightweight teamwork and gamification. Their results showed positive feedback from students, although they did not individually test the effect of each intervention as Campbell et al. did. Lacher and Lewis [6] tested the effect of pre-class video quizzes on student performance in a controlled study and discovered no significant effect. They postulate that while these quizzes before the class make students engage with the content, there is no deep learning taking place for most students, especially those with lower marks.

Computer Architecture was taught using a flipped classroom approach by Gehringer [7]. He taught both a graduate and undergraduate version of the same course and recorded the lectures in one for use in the other, to maximize reuse of content. In comparisons of student performance, the flipped class students fared worse. However, there were only 8 students and there were many differences between the classes so these results are unlikely to have much significance.

In this paper, an alternative approach to a Computer Architecture flipped class is presented, based on the positive lessons learnt, and with adaptations for a specific environment and cohort of students.

3 Outline of Computer Architecture Course

Computer Architecture is the study of computer hardware and the design decisions and choices that affect computer hardware. This course is studied either from the perspective of designers of hardware or designers of software. The former study architecture to understand how to design hardware while the latter study architecture to build better software systems.

This course was aimed at the latter group of students, in the second year of their degree, but included a mixture of Computer Science and Computer Engineering students in the classroom as the course was required for both degrees.

The content of the course followed the popular textbook by Patterson and Hennessy [9], with some contemporary modifications, such as the inclusion of an Open Hardware topic for relevance. The topics included in the course in 2015 were as follows:

- Introduction to computer architecture
- RISC and CISC CPU architectures
- MIPS Assembly language
- Pipelining
- Multicore CPUs
- Cache architectures
- Virtualization
- Performance and benchmarks
- Memory / SDRAM
- Secondary storage: hard drives, solid-state drives and RAID
- USB
- General purpose GPUs
- Open Source Hardware: Arduino
- Summary and concluding remarks

There were 14 classroom sessions in total. This is a very short course compared to offerings elsewhere so the focus of the course was on carefully selected high-level design concepts rather than the intricate details of any one section.

Each topic in the list above was the subject of a single classroom session in the course.

175 Students were enrolled in 2014 and 179 students were enrolled in 2015. The course was run in 2014 and 2015, with similar content and structure in both years. The following section describes the various elements of teaching and learning used in the classroom sessions.

4 Learning Design

4.1 Flipping on the Cheap

At the beginning of the course it was decided that this new approach would have to be low-cost. The flipped classroom approach, in general, requires a large financial and time investment, especially for the creation of lecture videos. In addition, any interactive experiential learning in class may require additional staff, such as tutors and teaching assistants where groupwork is needed.

Many universities simply do not have the resources available to conduct such experiments. There are typically no funds for producing pre-recorded videos. Also, students would be required to have access to written content such as a textbook and, realistically, not all students purchase these books. Further, it is not clear that such an approach will work for any given course so the risk factor is high and many universities cannot afford such risks. Finally, costs are divided into fixed costs, such as producing videos once for an entire group, and proportional costs, such as additional tutors to mark more assessments. For a course to be scalable, proportional costs must be managed so that flipping a classroom does not increase the cost of offering a particular course.

For the reasons above, it was decided that the cost of converting and running the course as a flipped classroom course needed to be minimal. Each element discussed below takes this cost reality into account in its design.

4.2 Content Videos

Every classroom session had an associated video clip that students were required to watch before the session. Each video clip was approximately 10 minutes in length. The videos were made available through a Youtube playlist [8], with links created on the Learning Management System (LMS) used at the university (see Figure 1). In addition, all videos were downloaded and made available for download via the LMS so students would not need external Internet bandwidth to access the videos.

In the particular case of Computer Architecture, there is a large number of videos from primary sources available online, such as an interview with John Hennessey on RISC. It was decided that these would be appropriate as primary sources of information. The first lecture was an overview of the course and this was recorded and put online.

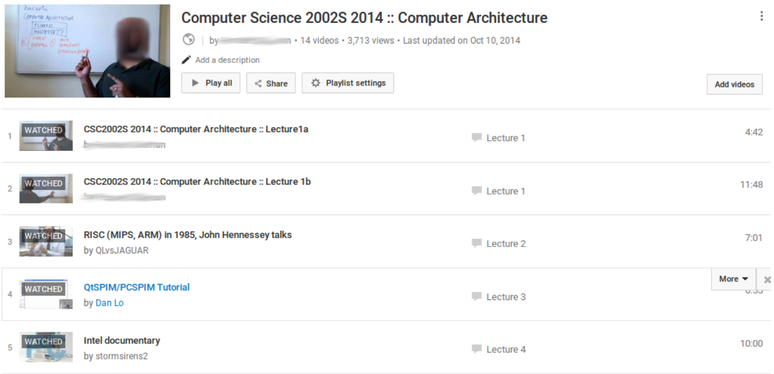

Fig. 1. Youtube playlist for class videos

The purpose of the videos was to create excitement about topics that might otherwise be considered dull. All of the videos were chosen because they focused on very specific and fundamental ideas students should be familiar with. For example, the video for the Performance topic was of a hardware reviewer demonstrating side-by-side timing of mobile phones running a popular benchmark suite. Students ought to identify with such examples more so than the theoretical benchmarking of hardware they do not have access to.

4.3 Readings

Videos by themselves do not contain sufficient technical information on which to base a discussion on Computer Architecture. Thus, it was decided that there would be accompanying readings. Some of the readings came from popular articles and extracts from the textbook [9] - the entire RISC/MIPS section had been made available for free by the publisher as the definitive guide to this topic. Most of the readings, however, were selected sections of articles from Wikipedia. Popular wisdom is that Wikipedia articles have poor quality, but some formal studies [10] have shown that the quality of Wikipedia articles is comparable to, for example, the carefully curated Enclopaedia Brittanica.

Wikipedia was selected in many instances because of many reasons, as outlined below:

- It is free and easily accessible on all devices.
- It is arguably the most current source of formal information in a rapidly-changing field like computer architecture.
- It is carefully checked and cross-referenced by the authors.
- Information can be either in-depth or span a breadth of subtopics.
- Links are provided to navigate to related topics (this is simply not possible in traditional textbooks).

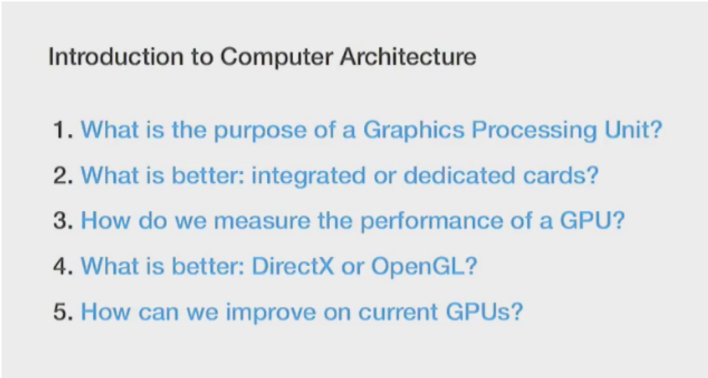

Fig. 2. Discussion questions for a session

4.4 Discussion

Each classroom session included a general discussion, facilitated by the lecturer. A number of questions were presented to the students and each was discussed in turn. Some questions were designed to highlight important concepts, others were to raise questions about the application of concepts while the final questions were often about a critical analysis of concepts.

Discussions were open-ended and could veer off into directions chosen by the students or lecturer. The lecturer served as more of a chair than a commentator. Some students were confused by this role, expecting the lecturer to provide the answers to all questions but, if that were to happen, there would be no real discussion within the class any longer.

Figure 2 shows a typical set of discussion questions.

4.5 Quizzes

Every classroom session included a quiz on the assigned reading and video. These were to ensure that all students did in fact read and view the assigned work before participating in the discussion. Students were given 5 multiple-choice questions on key facts from the assigned work. These were flashed on the screen in a timed manner such that each question and its options were only visible for 30 seconds.

The answers were filled in on a computer-readable form that could be scanned and automatically graded afterwards. This system was adapted for this course from one built for a previous course. No special equipment was needed - a common photocopier was used to scan the documents. The answer sheets (see Figure 3) were also produced on a photocopier or printing press, with varying skew and scale. Image processing automatically compensated for these issues, automatically generated a mark for each student, uploaded these marks to the LMS and made the answer sheets available to students as downloadable PDF files.

Most approaches used for in-class quizzes use mobile devices or laptops. However, those approaches require that the wireless network is operational and this

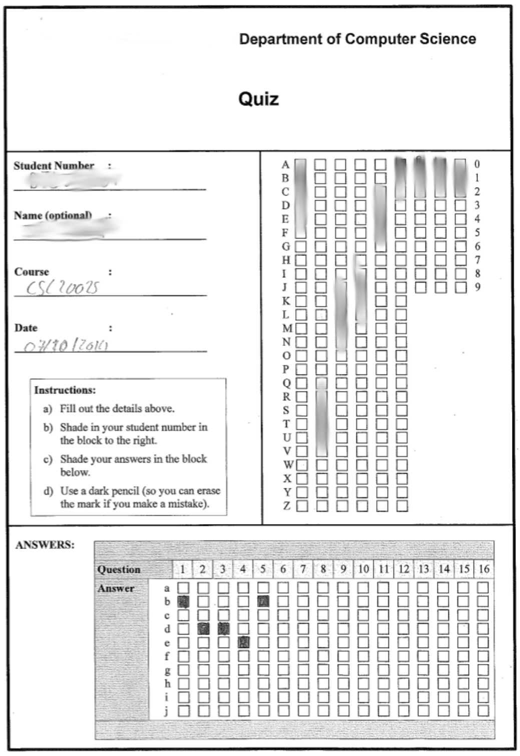

Fig. 3. Quiz answer sheet

cannot always be guaranteed. In addition, students could easily search for answers online if they are using a laptop. The integration of paper answer sheets into an otherwise digital workflow makes the system both scalable and robust, without increasing the technology requirements for students in the classroom.

4.6 Demonstrations

Demonstrations were included in almost all classroom sessions to create a stronger link between theoretical concepts and practical implementations. This was not, however, always possible because of some topics did not naturally lend themselves to realistic demonstrations e.g., pipelining. Many topics could easily be demonstrated, such as benchmarking software, virtualization (using VirtualBox) and open source hardware (using a Google Cardboard VR device). The remaining demonstrations were on current topics that were deemed useful to students at this level, such as the use of a Raspberry Pi computer.

Cost is always a factor in designing appropriate and feasible demonstrations. In order to demonstrate the Raspberry Pi in-class, a USB video frame grabber was used along with frame grabber preview software to simulate a display for

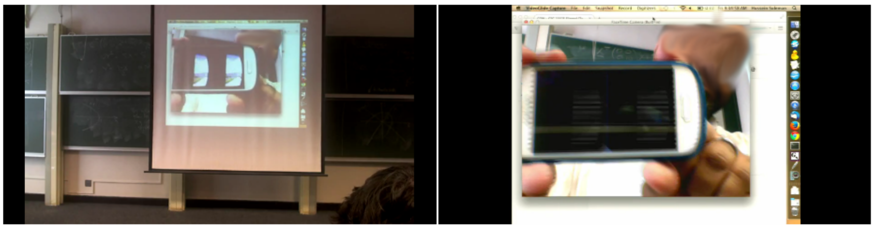

Fig. 4. Demonstration using laptop webcam

the device and have this reflect on the screen students were looking at. Figure 4 shows a mobile device being demonstrated to students using the lecturer's laptop webcam. These techniques allow for demonstrations with small devices to be used in a classroom without the need for additional staff - as long as the data projector's screen is viewable by all students, these demonstrations are scalable to the size of the class.

Finally, some demonstrations used software emulation of hardware devices. For example, a robotic arm that connected via USB would only work on an older computer so this was simulated using virtualization. This allowed for a wide range of hardware experiments to be conducted in class without having to haul additional equipment to the classroom.

5 Evaluation and Analysis

In all classes, students provide feedback as a matter of course. In this course, students were asked for additional feedback on elements of the flipped classroom model used. These, as well as the regular feedback, are discussed next.

5.1 Quantitative Feedback

In 2014, students were asked to indicate either a positive, negative or neutral reaction to each of the elements introduced in the course. Feedback was kept to a minimum because students were also providing feedback on the lecturer in a separate exercise.

Table 1 displays the summary of student responses.

Videos, demonstrations and discussions were considered to be mostly positive by students. It was expected that students would prefer to watch videos, given that students consume more video than ever before. Discussions were previously well received by students in 4th year, so students confirmed that even earlier in their degree they appreciate in-class discussions. Demonstrations got the most positive response, possibly because students were presented with a more practical view of the course than they would get in other courses.

Readings and quizzes were, as expected, not well received by students. Students were more negative about readings than positive, suggesting that the majority of students did not want to read before class. While many students were

Table 1. Student perceptions of aspects of flipped classroom experiment

	Readings	Videos	Quizzes	Demos	Discussions
Positive	19	46	31	58	49
Neutral	25	21	30	15	22
Negative	32	9	15	3	5

negative about quizzes, the results lean towards the positive. It may be that some students disliked the constant assessment (and the fact that quizzes were conducted at the beginning of the lecture) while others appreciated being forced to do the readings diligently.

This quantitative evaluation exercise was not repeated in 2015 because lectures were disrupted by student protests precisely as the course came to an end.

5.2 Qualitative Feedback

When asked for feedback on the experimental elements of the course, students were also asked for general comments. Some of the comments are listed below.

- "the theory was too much and it is difficult to understand without having done the things practically"
- "some pre lecture readings took quite a long time to go through the day before a lecture"
- "needs to give better notes and not just wikipedia pages"
- "I feel like no effort was put in with making our readings wikipedia pages ... And the fact that it was wikipedia was slightly off putting"

Students made similar comments in the lecturer evaluations in 2014 and 2015.

A major focus of the comments was the readings. Students made every conceivable argument against reading anything before class. The only argument that could be considered valid is that students have difficulty if English is not their first language. However, at an English-medium university, students more than halfway through a Bachelors degree ought to be able to read technical literature.

The elements of the course with positive quantitative feedback were not commented on by students.

5.3 Lecturer evaluation

Students in both years submitted lecturer evaluations for the course. In both cases, the numerical scores indicated that the lecturer was considered "above average" for effectiveness. This is in stark contrast to the written comments on how the readings were "horrible" or deemed completely inappropriate. When asked about the overall experience, students appeared to assess the overall experience but, when asked to provide general feedback, they only honed in on their issues with the readings.

This was an unexpected disconnect between evaluating the complete experience and evaluating a single aspect that had a profound impact on the students.

5.4 Analysis

From the various forms of evaluation and discussions with students, it is clear that students have an expectation that courses will be taught in a manner they are familiar with. In this case, students voiced strong opinions on how they think the course should be taught, what they think the content should be and how the content should be presented. They were asked if any of them had ever looked at any textbooks on the subject, and none of them ever had. These expectations are a cause for some concern, as it is clear that students are in a comfort zone from which they do not wish to be disturbed. The flipped classroom presentation of this course was more of a disruption than they had expected.

Reading is the biggest challenge for the future. Students made it clear that they do not read and do not wish to read. This is not a new challenge but Computer Science is changing and students cannot simply get a degree on the basis of mathematical skills - they now need soft skills like reading, writing and speaking or presenting. If students have difficulty in reading at second year level, maybe it is a sign that they need to do more reading from the beginning of the degree. Traditionally, the early part of the degree is focused on programming skills but maybe that needs to change.

Feedback from students also makes it clear that they have learnt particular ways of learning. A flipped classroom expects students to learn in a different way and students appear to have difficult in learning how to learn differently. This too may need to be addressed by exposing students to different ways of learning earlier rather than later in their studies.

Finally, students were given original unsolved problems to address in their final examinations. The vast majority of students were able to provide cogent arguments to support design decisions they would make in designing original computer hardware. Their specific answers, and a high level of achievement in general, suggest that students learnt not only the content that was expected but the critical thinking skills they were expected to learn. Thus, the goal of the course was achieved - students learnt precisely the skills they were supposed to learn. The only downside is they did not all enjoy the process of this learning. This outcome correlates with the study by Amresh, et al. [11], who also found that student performance in such a course improved but possibly at the expense of the student experience.

6 Conclusions and Future Work

This paper has reported on an experiment to use the flipped classroom metaphor in teaching a second year course in Computer Architecture. Many different elements of the course were changed to support this mode of teaching and do it without incurring additional costs. Some aspects were specifically chosen because of the nature of the specific course. The flipped classroom metaphor was mostly successful and addressed all the needs of the course, without increased cost. In future, this approach could be applied to various similar content-heavy courses or replicated to verify the outcomes and allow for deeper analyses.

Student feedback in various forms has indicated that students appreciated most aspects of the course, but were strongly opposed to reading. This is where the next level of intervention may need to take place. Writing across the curriculum [12], and indeed reading as well, may need to be adopted as a general strategy to train Computer Science graduates with new age skills that go beyond algorithms and logic. This will have the dual effect of creating graduates with broader skills and creating a student cohort that is able to learn effectively using a wider range of teaching and learning modalities. While it may be argued that these skills ought to already be in place by the time students arrive at university, it is clearly still a problem that needs to be addressed.

Acknowledgement. This research was partially funded by the National Research Foundation of South Africa (Grant numbers: 85470 and 88209) and University of Cape Town. The authors acknowledge that opinions, findings and conclusions or recommendations expressed in this publication are that of the authors, and that the NRF accepts no liability whatsoever in this regard.

References

1. Becker, W. E., Watts, M.: Chalk and Talk: A National Survey on Teaching Undergraduate Economics. The American Economic Review, 86(2), pp. 448–453, (1996). http://www.jstor.org/stable/2118168
2. Lage, M. J., Platt, G. J., Treglia, M.: Inverting the Classroom: A Gateway to Creating an Inclusive Learning Environment. Journal of Economic Education, 31(1), pp. 30–43, (2000). http://doi.org/10.2307/1183338
3. Bishop, J. L., Verleger, M. A.: The flipped classroom: A survey of the research. In Proceedings of 120th ASEE Annual Conference and Exposition, ASEE (2013).
4. Campbell, J., Horton, D., Craig, M., Gries, P.: Evaluating an inverted CS1. In Proceedings of the 45th ACM technical symposium on Computer science education (SIGCSE '14), pp. 307–312, ACM (2014). http://dx.doi.org/10.1145/2538862.2538943
5. Latulipe, C., Long, N. B., Seminario, C. E.: Structuring Flipped Classes with Lightweight Teams and Gamification. In Proceedings of the 46th ACM Technical Symposium on Computer Science Education (SIGCSE '15), pp. 392-397, ACM (2015). http://dx.doi.org/10.1145/2676723.2677240
6. Lacher, L. L., Lewis, M. C.: The Effectiveness of Video Quizzes in a Flipped Class. In Proceedings of the 46th ACM Technical Symposium on Computer Science Education (SIGCSE '15), pp. 224-228, ACM (2015). http://dx.doi.org/10.1145/2676723.2677302
7. Gehringer, E. F., Peddycord, B. W., III: The inverted-lecture model: a case study in computer architecture. In Proceeding of the 44th ACM technical symposium on Computer science education (SIGCSE '13), pp. 489-494, ACM (2013). http://dx.doi.org/10.1145/2445196.2445343
8. Computer Science 2002S 2015 :: Computer Architecture, https://www.youtube.com/playlist?list=PLElBQiaE5ZyFj2RLERYwH-vsF3lzEaM_h
9. Patterson, D. A., Hennessy, J. L.: Computer Organization and Design: The Hardware/Software Interface, 5th edition, Morgan Kaufmann (2014).

10. Giles, J.: Internet encyclopaedias go head to head. Nature, 428, 900-901 (2005). http://dx.doi.org/10.1038/438900a
11. Amresh, A., Carberry, A. R., Femiani, J.: Evaluating the effectiveness of flipped classrooms for teaching CS1. In Frontiers in Education Conference, pp. 733-735, IEEE (2013). http://dx.doi.org/10.1109/FIE.2013.6684923
12. Fell, H. J., Proulx, V. K., Casey, J. :Writing across the computer science curriculum. In Proceedings of the twenty-seventh SIGCSE technical symposium on Computer science education (SIGCSE '96), pp. 204-209, ACM (1996). http://dx.doi.org/10.1145/236452.236540