

LA-UR-15-24209

Approved for public release; distribution is unlimited.

Title: CS676 Real Time Systems Magnus Technology R&D Real Time Systems

Author(s): Nielsen, Roy S.

Intended for: Post on Academia.edu & researchgate.net

Issued: 2015-06-08

Disclaimer:

Los Alamos National Laboratory, an affirmative action/equal opportunity employer, is operated by the Los Alamos National Security, LLC for the National Nuclear Security Administration of the U.S. Department of Energy under contract DE-AC52-06NA25396. By approving this article, the publisher recognizes that the U.S. Government retains nonexclusive, royalty-free license to publish or reproduce the published form of this contribution, or to allow others to do so, for U.S. Government purposes. Los Alamos National Laboratory requests that the publisher identify this article as work performed under the auspices of the U.S. Department of Energy. Los Alamos National Laboratory strongly supports academic freedom and a researcher's right to publish; as an institution, however, the Laboratory does not endorse the viewpoint of a publication or guarantee its technical correctness.

CS676 Real Time Systems
Magnus Technology R&D Real Time Systems

Roy Nielsen

May 17, 2017

Table of Contents

Real Time System Overview and Design	2
Magnus Technology R&D Organization	2
Real Time System Background	2
Main Manufacturing Tool	3
New Project Requirements	3
Real-Time System Properties	5
Properties of a Real-Time System	5
Real-Time Concurrency Examples	5
General Resource Management	7
Real-Time Resource Management	7
Schedulability	8
Concurrency Mechanisms Problems and Solutions	10
System Complexities	10
Movement Calculation Optimization	10
Recommended Prioritization, Budgeting and Implementing Strategies	12
RTS Implementation and Evaluation	13
Approaches for Optimization	13
Ensure Redundant Hardware Systems	14
Moving Beyond the Pipeline	14
Dynamic Programming	14
Technology New to the FirePick Delta	15
Conclusions	16
References	17

Real Time System Overview and Design

Magnus Technology R&D Organization

Magnus Technology R&D is a research and development organization whose goal is to open source and advance technology. The purpose being to create something that makes lives better. The primary avenue for these activities is 3D printing and related technologies. While the organization uses several different devices, the FirePick Delta(Tin Whiskers Technology, 2015; Nielsen, 2015a) will be the principle one used for development.

The company has completed several projects and is taking on a new one which will now be its primary focus. This will be to research, prototype, develop, and construct motors that are entirely manufactured with the FirePick Delta. 3D printing technology has advanced to the point where there are printing supplies with metallic, magnetic or conductive properties. These are the ingredients of a motor.

Real Time System Background

Any computer or system can be considered a real time system. For example, a word processor takes input and processes the information to perform some kind of output, such as sending a job to the printer. Many devices take a set of inputs, process the information and perform an output task. Control systems at a nuclear power plant, the brake system in your car, and the the control system on an airplane are more examples of systems that require a timely response. These are categorized under one of three definitions of a “real time system”. How the real time system responds to inputs by performing task outputs falls into one of three categories. The first two, hard and soft, are spoken of by Oshana above. The third category is known as “firm”, and falls between hard and soft(Srinivasan, Pather, Hill, Ansari, & Niehaus, n.d.).

Failure in a “hard” real time system is defined by catastrophic failure. In this case, timing is crucial to system success. Safety critical systems such as controls in a nuclear plant, air plane controls, and some medical devices fall in this category. A “soft” real time failure just means a delay in execution of tasks instead of catastrophic failure. An example of a “soft” real time system might be a word processor’s interaction with the operating system. Failure in a “firm” real time system has “hard” real time system timing requirements while taking the “soft” penalty of a delay of execution rather than catastrophic failure. The three categories of real time systems can describe just about any computerized business practice or computer system.

Main Manufacturing Tool

The FirePick Delta is a microfactory in itself. It is a versatile manufacturing device that can switch heads to perform different functions. The initial functionality for the machine is 3D printing, laser engraving, and pick and place.

Most people are familiar with the first two, but not pick and place. Pick and place is the act of picking up electronic parts and placing them correctly to construct a printed circuit board. Processes the FirePick Delta may be capable of in the future could include milling of traces, solder paste deposition as well as soldering of the parts once they are placed.

These tasks are mutually exclusive as the head needs to be switched out to perform each different function. The 3D printing functionality will start with a single head with a single feed, to be switched out for printing in different colors. The laser engraver is not intended for 3D prints, but for wood projects. The pick and place head will have a vacuum that will pick up electronic parts and place them on a circuit board.

These tasks seem simple until they are implemented. For instance, pick and place requires a downward looking camera for correct placement. It will also require an upward looking camera to make sure the legs of the parts are in the right orientation for assembly. These inputs will be processed by the embedded real time device to control the movement motors as well as the vacuum motor holding the part.

New Project Requirements

For home users, the FirePick Delta requires only soft real time functionality required for 3D printing. To be able to prototype and manufacture motors will need to use multiple functions of the machine. Both the 3D printing and pick and place functionality will be needed to complete the motor. The FirePick Delta may need to be modified to handle assembly of the parts.

OpenCV will be used to handle the image processing. People have committed to use one of several embedded boards to control the FirePick Delta. The Arduino Uno, Raspberry Pi 1 & 2 are among the embedded devices currently used. The embedded board connects to the motor control board that is specifically designed for this delta device.

When motor prototypes are proven viable, the motor will be commercialized. This process will require a higher level of real time system, as the business will depend on a faster responding system. Manufacturing speed could make or break the business, and the

faster the machine has to perform, requiring higher hardware and timing specifications.

The FirePick Delta is critical to both motor prototyping and commercial manufacturing due to its versatility.

Real-Time System Properties

Properties of a Real-Time System

Properties of a real time system include:

- Reliability - the FirePick Delta mean time to failure is a very long period of time
- Reproducibility - the ability to consistently repeat the same job with the same results
- Integrity - the parts of the device are solid and able to take the vibrations of the motors without falling apart
- Maintainability - the ease with which the device is maintained
- Concurrency - being able to control four or more motors at once based on multiple sensor inputs

For entry level machines targeted to the home user, the parameters above need not very restrictive, leading to less capable hardware that is less expensive to enable easy entry into the field. Business will require more from the machine so these attributes will need to be well defined with much higher performance required for the control hardware, to be first to market as well as best in the market.

Real-Time Concurrency Examples

A real time system collects information from outputs, processes that data and, based on the results, performs some output in real time. This kind of system will often have to perform several tasks concurrently(Kagstrom, Tuska, Grahn, & Lundberg, n.d.). Three methods of describing this problem include multitasking, multi-threading and multiprocessing. Multitasking is the act of handling multiple tasks at the OS layer. Multi-threading is the act of the kernel handling multiple tasks and multiprocessing the act of multiple processors handling data.

The FirePick Delta is a device that has multiple inputs with an embedded device processing the data to control multiple outputs. The device is being designed to perform multiple tasks, which include 3d printing, pick and place and laser engraving and future it may be adapted to additional tasks as well. Once the FirePick Delta leaves the beta process and goes into production it will be a firm real time system, as it needs accurate timing to respond to inputs. Initial production 3D print speeds will be average for machines in the home user and small business realm. Currently it takes a week to weeks to get a small run of boards back from a manufacturing shop. The FirePick Delta will bring

that time down to a matter of days instead of weeks with its pick and place functionality. Laser engraving will likely start production being as fast as 3d printing movements.

It will have a collar, or effector held by the robot arms to hold the head doing the work such as the 3D print head, vacuum head for pick and place, or laser head for engraving. The machine has an automated head changer to automatically switch heads between tasks.

This delta printer will have several inputs. One is a temperature sensor on the 3D print head to make sure the print filament is being melted at a consistent rate. It will also have two cameras, one pointing up from the base, and one looking down from the effector. Several different kinds of feeders will be available for the device to feed parts to the head for placement on the circuit board. Some of the feeders may be automated. The FirePick Delta will need to wait for the feeder to have a part ready before it can pick up a part. With a varying number and complexity of inputs, the controller needs to be flexible.

A Cartesian based printer (Campbell, 2015) such as the Ultimaker model moves the motors only on linear Cartesian planes. A generic delta printer moves multiple motors in non-linear ways to map onto a Cartesian coordinate system. In addition, the FirePick Delta needs to control several functions simultaneously. For 3D printing, this includes the three motors for movement, as well as the heater for extrusion, motor for extruding the filament and heating the print bed. Other functions that need control based on feedback include solder paste delivery, part feeders, altering vacuum pressure and camera setting as well as laser head control.

One of the major goals of the FirePick Delta is to make the device as user friendly as possible. One way to do this is to abstract and hide the complexity in software and firmware. An example of this is to not require careful alignment of a printed circuit board on the machine platform, but allow semi-random placement of the board. To do this, pick and place will require image processing for feedback for motor control. This may require a multi-processor embedded system. The embedded device needs to process images to not only move the arms for the part location on the board, but also to determine the board orientation. One plan is process an image of the board and use three or more holes as reference points for movement. This will allow for a user to put the board on the machine at a semi-random orientation on the platform for pick and place. This will remove the need for strict placement on the device platform.

One of the target audiences for the FirePick Delta is a company that requires small runs of boards with cases to be prototyped quickly. The FirePick Delta can also perform rework including surface mount parts on the boards. Although being able to execute prototyping in-house greatly reduces time to market, being able to print boards in minutes instead of hours is a goal for some of the engineers for the machine. Fast, accurate prototyping can mean the difference between business success and business failure.

General Resource Management

Resource management for the FirePick Delta can vary greatly from an entry level machine to a faster, more precise, professional version of the machine. Entry level 3D printing and pick and place will have standard speeds equivalent to other entry level kit machines. For instance the entry level FirePick Delta 3D print functionality will be similar to the speed of Cartesian or delta printers(makezine.com, 2013). A professional machine will perform faster than the base model. Faster head movement means faster calculations, which means a faster, perhaps multiprocessor board. Businesses rely on the speed and accuracy and are willing to pay a premium to obtain these. Beating a competitor to market can mean the difference between business success and business failure. For entry level machines, resource management and hardware requirements can be more lax.

Real-Time Resource Management

Resource management is a complex endeavor for the FirePick Delta. For 3D printing purposes, three motors need to be managed for movement of the 3D print head and another motor for managing the feed rate of the filament being used as the additive, in a single filament head. There also needs to be independent control of the temperature of the head based on a temperature sensor inserted into the head. This printer has a different arm movement mechanism than the Rostock model printer(Boots Industries, n.d.), making calculations for radial movement rather than linear. This difference can make the firmware calculations for movement complicated, requiring faster hardware for calculating distance and direction. Depending on the hardware configuration, controlling these parameters may need to be more finely tuned than other configurations.

Entry level machines need to be as inexpensive as possible, which determines the specifications for the hardware. The Linux OS is what runs the FirePick Delta. If the Linux host is also running X11 to provide a graphics interface to a TFT interface, that will take much more memory than a twenty digit seven segment LED display. Of course the LED display will run on lower spec hardware, which will be less expensive.

The more features the machine has and faster the machine is required to perform, the higher the hardware requirements need to be. An additional feature that is being planned is inductive heating for extrusion of the filament. This concept was recently introduced to the FirePick Delta Google Group(Nielsen, 2015b). The CPU would need to be capable of calculating and generating the variable signals to the inductive coil for the heating process.

Faster direction, velocity, and acceleration calculations for radial movement of each of the three printer head movement motors will require a faster and more tightly controlled

hardware environment.

Pick and place functionality will perform in a fashion similar to 3D printing. However, instead of temperature control and filament feed speed, the head will need rotational orientation control as well as vacuum control. The hardware requirements for this pick and place will be based on the image processing for both the upward facing and downward facing cameras. Again, as above, a slower less expensive entry level machine could have much lower requirements for hardware than a more expensive, faster, professional version of the machine that needs to perform at much higher levels.

Schedulability

Schedulability analysis(Gujarati, Cerqueira, & Brandenburg, n.d.) is the mathematical comparison of different process scheduling mechanisms for a particular system. There are general algorithms and methods for determining scheduling mechanisms, however the FirePick Delta team has not settled on a hardware platform for controlling the system. Atom, X86, FPGAs and Arm boards are among the contenders. Single and multiple processor systems are being considered, widening the range of possible algorithms to use for schedulability analysis. The boards under consideration have a wide range of frequencies which will also affect the analysis. The slower the system, the tighter and stricter the scheduling will need to be.

Static and dynamic methods can be used in scheduling processes on a system(Andersson, Baruah, & Jonsson, 2001a). Static scheduling gives a weighted priority to tasks on a system at compile time. The FirePick Delta could use some static scheduling to perform certain tasks such as stopping the motors when a print or pick and place boundary is met. If a motor is not given a signal that a boundary is met, the motors could continue attempting to move past the end point and strip gears or break joints. These end points could be compiled into the system being given high priority thread.

Thread sensitive scheduling leverages the hardware threads. The FirePick Delta project has not proven its guidance and visualization algorithms on the lower speed, less capable processors. Control boards with faster thread capable processors are being developed. Video and picture analysis is needed to control four or more motors. Thread sensitive scheduling on processors that implement hardware threads gives the more capable processors an advantage.

The pick and place control relies on processing visual cues to control both motors and vacuum. There are two cameras on the FirePick Delta, one looking up from below and one looking down. The camera looking up must determine if a component is in the right pin orientation and the one looking down must be able to put the part in the right place. The vacuum is used for picking up a component and turned off when placing the component. The FirePick Delta must be able to sense if it is holding a component, so a

vacuum presser sensor is necessary. A motor is also required to turn the part it is holding to orient it correctly as it is being placed.

Monotonic process scheduling may significantly slow down the FirePick Delta as only one process is scheduled at a time with this method. Thread sensitive scheduling would give the FirePick Delta an advantage to processing all the required inputs and controlling the outputs in a timely manner.

Concurrency Mechanisms Problems and Solutions

System Complexities

The most complicated processes for the FirePick Delta are the movement calculations for the three robot arms. The movement has to be completed quickly enough for all three motors to appear to move at once. Movement for the FirePick Delta can get complicated. On the firepick@googlegroups.com mailing list, several methods were suggested to calculate movement. One was to look at the closest motor movement point in every direction and determine which was closest to the destination point. Another developer suggested 180 degrees of the calculations be performed and then determine the direction closest to the destination point. I have developed a dynamic programming method to simplify determining 3D print head movement. These calculations need to take place in such a way as to move all three motors at once.

Movement Calculation Optimization

Instructor Jha at St. Cloud State University in his Design and Analysis of Algorithms course gave a good description of dynamic programming and its relationship to optimality.

”Dynamic programming relies on an optimal decision sequence [that] must consist of optimal decision subsequences.” (P. K. Jha, n.d.)

A popular algorithm that illustrates dynamic programming and optimality is the calculation of the edit distance (Jurafsky, n.d.) between two strings to determine a pattern match. In calculating the edit distance, the algorithm remembers past optimal calculations to help determine the optimal path to a pattern match.

Movement Calculation for 3D printing

The FirePick Delta 3D printing could use dynamic programming to determine optimality from point to point. The possible combinations for movement for each motor are forward, backward and staying in place. There are three motors with three combinations for movement. This would make nine possible directions to go from the current location. In determining the next motor movement step to take, an algorithm can check each combination of motor movements to calculate the closest point to the destination point. Once the motor combination has been calculated, the device needs to send the motor control commands to the motor controllers. If a piece of specialized hardware is added, one

of eight numbers can be sent to the new circuit. The new circuit can simultaneously send move commands to the three different motor controllers, much the same way a seven segment LED driver chip works. Offloading the serial nature of the motor control from the software to hardware will greatly simplify what the operating system has to accomplish.

Or, dynamic programming could be used to find an optimal path. Using vector Cartesian math, the slope and direction from the current point to the destination point can be calculated. Since this calculation would be only for a plane, and not three dimensional, one would need to hold one plane constant and compare the line along the third axis. Instead of calculating each motor step to the destination, the print head continues along the calculated slope until reaching its destination. Once reaching the destination point, it is time to re-calculate direction and magnitude to the next destination point.

Dynamic programming can simplify the calculations that need to be made for motor control. This, along with offloading actual motor control, will allow for simpler, less expensive hardware to handle movement calculations.

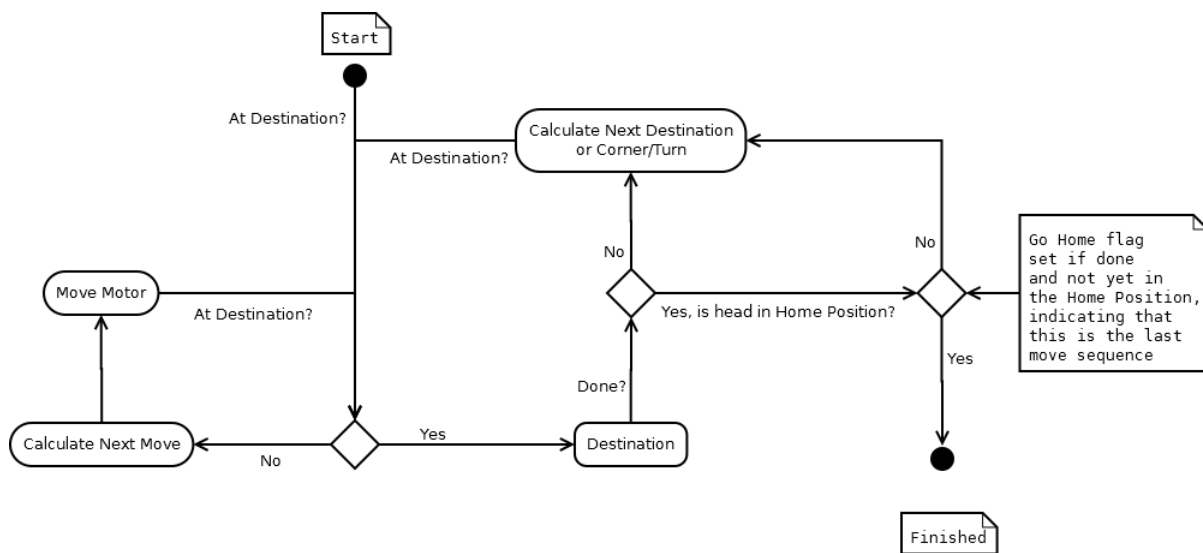


Figure 1: Print Head Movement State Diagram

Movement Calculation for a Circuit Board Build

When considering the path calculations for pick and place for the FirePick Delta, movement calculations need to include obstacle avoidance. The controller would need to take the positions of components already placed into account and either go over or around them to avoid collision. Instead of weaving around components to place the current part, the simpler task would be to go over the components already in place. Once over the target location, the part could be turned to the correct orientation and then placed. As dynamic programming remembers critical past calculation results, the location of each already

placed component could be stored in a lookup table for path calculation purposes. The path between the part pick up and drop locations could be broken down to points just above already placed parts. Then movement calculations could be made as above in the 3D printing example.

Recommended Prioritization, Budgeting and Implementing Strategies

When the overall FirePick Delta control is examined, rather than just the movement, one will need to consider scheduling paging between processes to ensure that enough processor cycles get provided to each process to complete the task. For instance, when 3D printing, one needs the filament feeder to feed the filament to the print head at a rate that will extrude just the right amount of filament for the movement taken. If the movement motors and extrusion motor get out of sync, there will be too much or too little filament being extruded, ruining the print job. Part of the filament extrusion process is sensing the temperature of the print head and controlling the output temperatures as well. For pick and place, the vacuum and rotational motors will need to be moved at the appropriate times, so proper scheduling needs to be performed in this arena as well.

For faster processing, CPUs with multiple hardware threads or multiple CPUs will make concurrent processing an essential component for a faster 3D print or circuit board build.

RTS Implementation and Evaluation

Magus Technology R&D currently has what is known as a pipeline methodology for prototyping systems. A pipeline process is serial in nature; one step after the other until a prototype is built. The steps that create a prototype include:

- 3D printing a case and chassis
- Circuit board milling, sometimes manual etching
- Solder paste deposition
- Pick and place of parts
- Soldering, most often done manually
- Assembly of parts

All of the steps can contain some level of manual work and most steps need to take place in a specific order. The case and chassis of the system can be manufactured any time before the assembly happens. Once one prototype build finishes the first stage and starts the second stage, another prototype build can start the first, always keeping each stage in the process busy. Problems occur when a machine goes down as the work then must be done manually. The less expensive machines may have a whole machine in storage as backup to swap out a broken one. The more expensive machines have some critical spare parts, but it takes time to replace these, meanwhile work proceeds manually. Magnus Technology R&D would like to get to a point where it has multiple redundancy in its process as Zorin and Kostenko (Zorin & Kostenko, 2015) describe.

The company has some forms of multiple redundancy, and is looking to increase optimization of systems so it can handle more business.

Approaches for Optimization

As an outside systems analyst, I recommend four targets for optimization of manufacturing processes.

- Ensure redundant hardware systems
- Move beyond the pipeline
- Use dynamic programming for path navigation and performing work
- Develop and integrate new technology into the FirePick Delta

Ensure Redundant Hardware Systems

The FirePick Delta has been designed to be modular and flexible. The design is such that it can be controlled with an Arduino, Raspberry Pi or SmoothieBoard. The actual motor controllers are on separate mini-cards that are plugged into the controller board. The design is very modular to allow for quick replacement of parts so a machine can immediately be put back into service. All of the source files for the hardware are open source and can be retrieved at any time for people to use to modify designs for their own needs. The heads are easily replaced and the system is automated so that the machine has access to up to four heads at a time. Parts can be modified and manufactured with quick turn around to create a larger or smaller machine, or even be used for a new purpose such as pipetting for bio-chemical processes.

Moving Beyond the Pipeline

This machine has been designed such that it can be a microfactory itself, so many of the other machines being used for prototyping can be retired. When the FirePick Delta is deployed, rather than implementing a pipeline, it will be treated more like a super computer cluster. A cluster based job scheduler can determine what machine resources are available and send jobs to those assets as determined by the general job scheduler algorithm. When all the FirePick Deltas are in working order, all their multi-function resources are available for use.

If the laser head etching head is broken, the scheduler could continue process jobs for that specific FirePick Delta regarding pick and place as well as 3D printing. The scheduler would just remove the laser etcher capability from the pool until there is an opportunity for it to be fixed.

As one process might cause a bottleneck in the cluster instance, more machines could be assigned that specific task to relieve the backlog. When the backlog is satisfied, the job scheduler would re-balance the jobs to better fit the available resources.

Dynamic Programming

Dynamic programming for the purpose of head movement was described in the Concurrency Mechanisms Problems and Solutions section. Another application of dynamic program could be the control of an inductive heating of the filament.

The inductive heating would include pulse width modulation(PWM) of one to over four hundred kilohertz waves depending on the design of the coil(Zinn & Semiatin, 1988). The width and height of the signals would be controlled by digital capacitors and resistors. The heater needs to be kept at a fairly constant temperature and needs to function in a variety of different environments, so one specific set of settings might not work near the

ocean versus high mountain desert. Additionally, different materials will react differently to inductive heating.

This will be especially evident with the use of magnetic, conductive and metallic materials that will be used for printing the 3d motor. Due to environmental conditions and different material properties, creating a reliable table for every condition is unmanageable. A self adjusting algorithm will use dynamic programming to control the PWM. This will lead to more efficient use of energy for heating the filament.

Technology New to the FirePick Delta

A new technology to the FirePick Delta will be a touch screen interface that also shows a model of the task it is currently performing as the process proceeds. This could also be provided over a web interface at different video resolutions. It would also give an indication of a process that might be going wrong, so it could be interrupted and restarted. Adding this 3D modeling capability would require more processing power of the controller, increasing the cost of the machine. This capability would minimize the cost of loss of materials during the manufacturing process, which would be especially important to small businesses and research organizations.

Conclusions

1

The FirePick Delta is a tool that offers more flexibility than employing several individual machines of varying types that perform separate functions. Having a cluster of machines that perform several functions can lead to optimized manufacturing time, giving the owners more availability for additional projects.

The ability to construct 3D print motors with the FirePick Delta will give the advantage of 3D printing an entire drone. This would be advantageous for several organizations including:

- search and rescue
- emergency response
- crisis management
- recovery from a natural disaster

There are other uses such as law enforcement and military uses. Research and development at universities could also benefit from the ability to entirely print drones.

The FirePick Delta has a vibrant and active development community that continues to add to and improve the device.

¹Some editing suggestions made by my wife Eileen Nielsen were integrated after the class was over.

References

- Andersson, B., Baruah, S., & Jonsson, J. (2001a). *Static-priority scheduling on multiprocessors* (Tech. Rep.). IEEE Real-Time Systems Symposium. Retrieved May 17, 2015, from <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.21.7889>
- Andersson, B., Baruah, S., & Jonsson, J. (2001b). *Static-priority scheduling on multiprocessors* (Tech. Rep.). 22nd IEEE Real-Time Systems Symposium. Retrieved May 4, 2015, from <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.21.7889> doi: 10.1.1.21.7889
- Boots Industries. (n.d.). *Rostock 3d printer - bi edition: Getting started guide* (Tech. Rep. No. 1.0). Retrieved May 8, 2015, from <https://www.bootsindustries.com/downloads/Rostock%20BI%20Manual.pdf>
- Campbell, C. (2015, March 10). *Cartesian, delta, and polar: The most common 3d printers* (Tech. Rep.). Make:, Maker Media Inc. Retrieved April 21, 2015, from <http://makezine.com/2015/03/10/cartesian-delta-polar-common-3d-printers/>
- Gujarati, A., Cerqueira, F., & Brandenburg, B. B. (n.d.). *Schedulability analysis of the linux push and pull scheduler with arbitrary processor affinities* (Tech. Rep.). Retrieved May 17, 2015, from http://www.mpi-sws.org/~arpanbg/papers_pdf/ecrts2013_paper.pdf
- Jurafsky, D. (n.d.). *Minimum edit distance* (Tech. Rep.). Stanford University. Natural Language Processing course. Retrieved May 10, 2015, from <https://web.stanford.edu/class/cs124/lec/med.pdf>
- Kagstrom, S., Tuska, B., Grahn, H., & Lundberg, L. (n.d.). *Implementation issues and evolution of a multiprocessor operating system port* (Tech. Rep.). Blekinge Institute of Technology, School of Engineering, Department of Systems and Software Engineering.
- makezine.com. (2013). *make: 3d printer buying guide, the ultimate guide to 3d printing* (Tech. Rep.). Author. Retrieved May 8, 2015, from http://www.timelab.org/sites/all/files/MAKE_Ultimate_Guide_to_3D_Printing.pdf
- Nielsen, R. (2015a, February 17). *New generator technology 3d printed generator prototype darpa phase i systems engineering management plan(semb)* (Tech. Rep.). Written about a Fictional company for CS672 Systems Engineering Methods course.

- Retrieved April 20, 2015, from
https://www.academia.edu/11375014/New_Generator_Technology
- Nielsen, R. (2015b, May 8). *subject: fast controller board* (Tech. Rep.).
 firepick@googlegroups.com. Retrieved May 8, 2015, from
<https://groups.google.com/forum/#!topic/firepick/171IFF3Y3sU>
- Oshana, R. (n.d.). *Introduction to embedded and real-time systems* (Tech. Rep.). Texas
 Instruments. Retrieved April 20, 2015, from
http://verse.aasemoon.com/images/8/8f/Embedded_Realtime.pdf
- P. K. Jha. (n.d.). *An introduction to dynamic programming* (Tech. Rep.). St. Cloud State
 University. CSCI 404/504: Design and Analysis of Algorithms course. Retrieved May
 10, 2015, from <http://web.stcloudstate.edu/pkjha/CSCI504/DynamicP.pdf>
- Srinivasan, B., Pather, S., Hill, R., Ansari, F., & Niehaus, D. (n.d.). *A firm real-time
 system implementation using commercial off-the-shelf hardware and free software*
 (Tech. Rep.). Electrical Engineering and Computer Science Department, University
 of Kansas. Retrieved April 21, 2015, from [http://citeseerx.ist.psu.edu/
 viewdoc/download?doi=10.1.1.138.449&rep=rep1&type=pdf](http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.138.449&rep=rep1&type=pdf)
- Tin Whiskers Technology. (2015). *Firepick delta introduction* (Tech. Rep.). Tin Whiskers
 Technology, FirePick Services and open source developers. Retrieved April 20, 2015,
 from <http://delta.firepick.org/introduction>
- Zinn, S., & Semiatin, S. L. (1988). Coil design and fabrication: part 3, fabrication
 principles. *Heat Treating*. Retrieved May 17, 2015, from
<http://www.stanleyzinn.com/pdf/coil.design.pdf>
- Zorin, D. A., & Kostenko, V. A. (2015). Algorithm for synthesis of real-time systems under
 reliability constraints. *Journal of Computer & Systems Sciences International*(51(3)),
 410417. doi: <http://dx.doi.org/10.1134/S106423071202013X>