Autonomous Multi-cycle Farming in Space

Team Members:

Name	**Contact**
Client & Sponsor: Philip Chan PhD	pkc@cs.fit.edu
Christopher Milsap	cmillsap2013@my.fit.edu
Giampiero Corsbie	gcorsbie2018@my.fit.edu

Project Plan v1.5
Created: 1/15/20
Revised: 1/22/20

Client Meetings:

Date	Reason
January 17, 2020	Initial Meeting
Starting Jan 31th, 2020	Meeting every two weeks

Project Plan Version Control:

Document ID: PP-000001				
Revised Version	**Date**	**Collaborators**	**Version**	**Reason**
N/A	1/15/2020	All	v1.0	Initial creation
1.0	1/22/20	Chris	v1.5	Milestones and diagram changes

[1]

Document Reference:

Log of all document references where version number is needed. To maintain version control.

Document	Date Refenced	Version Referenced

Project Definitions:

Word/Phrase	Re-worded/*Notes*:	Definition
Autonomous Multi-cycle Farming Device	The AMCFD	The subject of this project and the product being created
Control(s)	*Not re-worded & definition may not always apply to*	The process by which software dictates the

[1] When updating this document's version update the version found on the title page, name, referenced documents, baseline and footer ↘

	software	behaviour and movement of certain devices or software triggers.
Software Abstraction	*Not re-worded*	A high-level description of implementation instead of giving detailed code. The software abstraction column is given to show where computer science engages with project features
Embedded, microcontroller, single board computer	*May be used interchangeably & not re-worded*	The computer chosen to be the brains of the AMCFD. It will manage the various components of the AMCFD

Motivation and Goals

In preparing long lasting missions in space, one of the most important factors for the well being of an astronaut is ensuring they are healthy and well fed. Given the cost of launching materials and the busy schedules of crew members, the optimal way of producing food to meet nutritional needs is through an automated farming system.

This system must complete multiple autonomous cycles from seed to harvest, then cleanup. To ensure astronauts' longevity and health, it must be highly reliable and use few resources. Given this will be initially designed to be aboard the International Space Station, it must fit within one of the standard experimental lockers aboard the ISS; the ultimate goal is to provide a renewable food source for astronauts. The goals for this project can be broken up into ultimate and current, where the ultimate goals are future oriented with how/what we project the device to meet the needs of its users. The current goals define a scope for this senior design project and a few project iterations after. Current goals for this project include high crop yield in accordance with what NASA produces, scheduling stability where the tasks of the device are completed correctly; on time, in expected order, & efficiently. Also improved crew morale and UX/UI study compared to current software at NASA.

The computer science goals on this project will be to provide the requirements and software needed to automate various parts of the AMCFD. These software needs can be categorized into sensor reading, actuator controls, timers, scheduling, interfaces, operating system components, and various resource management i.e. mitigating power drawn and pushed/pulled data.

As an extension we will build out a software system where the users can access prewritten configurations for the device in a configuration library. The application will provide configuration tools, several reports for the users so that efficiency and correctness can be monitored, and manual control of the device.

To understand the goals of this project we need to define the needs listed above.

- **Sensor reading:** Data from sensors that will be attached to the AMCFD to be used by the the AMCFD to make decisions at each phase of the cycle.
- **Actuator controls:** The software mechanism to move and command various motors "actuators" to complete certain tasks.
- **Timers:** Timing is extremely important in farming and especially important in space farming; all the sensors and actuators must read and perform at the right time in order for the automation process to cycle properly. Timers will run over dynamic tasks which will be periodic actions based on a delay between the termination of one task, the commencement of the next, and/or other triggers, along with giving timing offsets from dynamic tasks to the driver.
- **Scheduling:** Highly productive people need schedules to initiate and complete their task lists, it is the same with a computer. In order to make sure everything cycles correctly and completely the AMCFD must follow a schedule which will be determined by software. The driver will keep a list of times at which cycled tasks are to take place. Scheduling will also be a component of the operating system on the chosen microcontroller.
- **Interfaces:** The AMCFD will be fixed in an experiment locker on the ISS that will require ISS and AMCFD programs to talk to one another.
- **Operating system components:** Timers and scheduling will also take part in the operating system of the chosen embedded board.
- **Resource Management:** Resources in space are limited and extremely valuable. The AMCFD will be fed power from within the experiment locker along with push and pull requests[2] from the interfaces that need to be metered and controlled.
- **UX/UI:** User interface and user experience.

[2] Push and pull requests are basic elements of two-way communication between machines. If machine A has data to send it will push it to machine B and if machine B needs data it can pull it from machine A.

Approach - Phases

To ensure the wellbeing of the astronauts the system will use single board computers and/or microcontrollers. Sensors will be powered on at appropriate intervals to conserve power. Once the sensors detect the farm is ready to move on to the next stage, various physical systems will be engaged to transition. These stages are described below and each action of the AMCFD has a corresponding software abstraction.

Seed:

Action	Software Abstraction
Planting seeds	Scheduling and timing actuators to plant the seeds while reading in data from the AMCFD's internal environment
Detecting germination	Reading in sensor data to detect that the plants germinate properly

Grow:

Action	Software Abstraction
Providing a nourishing environment by sensing environmental factors	Using sensors to detect environmental compositions like Nitrogen and CO_2 and to trigger appropriate system responses via controls using reactive autonomy
Tracking plant growth phases over time	Using sensors to detect changes in plant growth over time and cataloging those changes in a database to make any adjustments to the actuators if needed using proactive autonomy
Water needs	Tracking/sensing water levels, and controlling water pumps to provide water to plants as needed

Lighting needs for photosynthesis	Turning light(s) on and off
Detecting diseased or under-grown plants	Using artificial intelligence[3] to monitor for potential diseased plants based on visual data

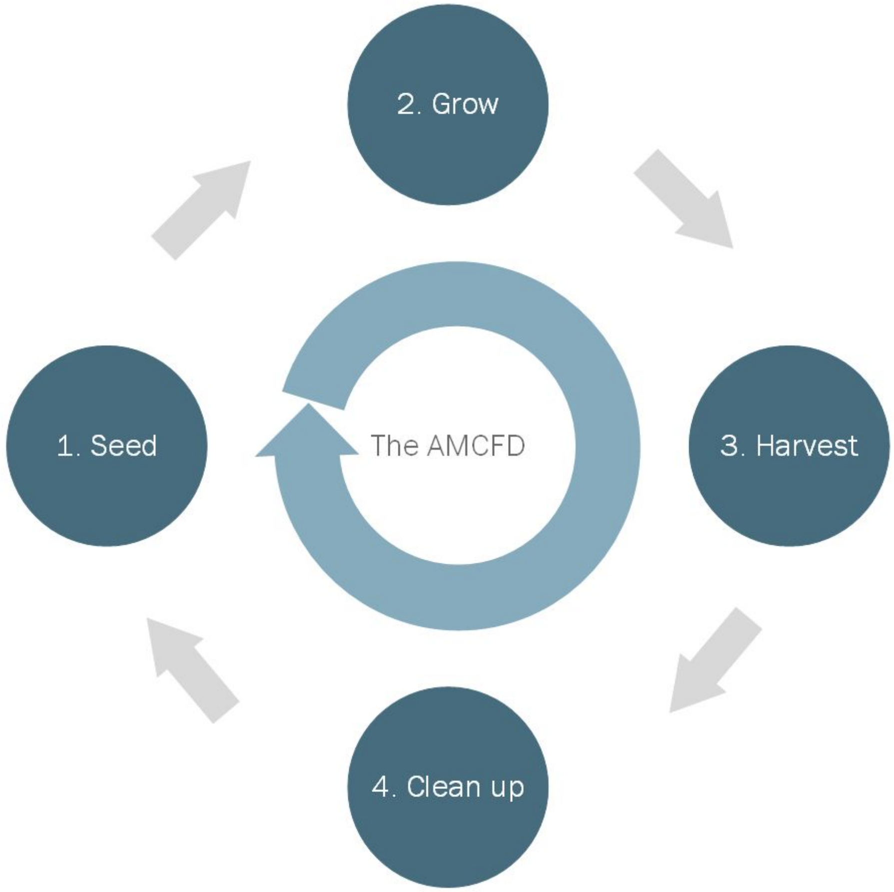
Harvest:

Action	Software Abstraction
Remove the grown plants for consumption	None, this will be done by the astronauts

Clean up:

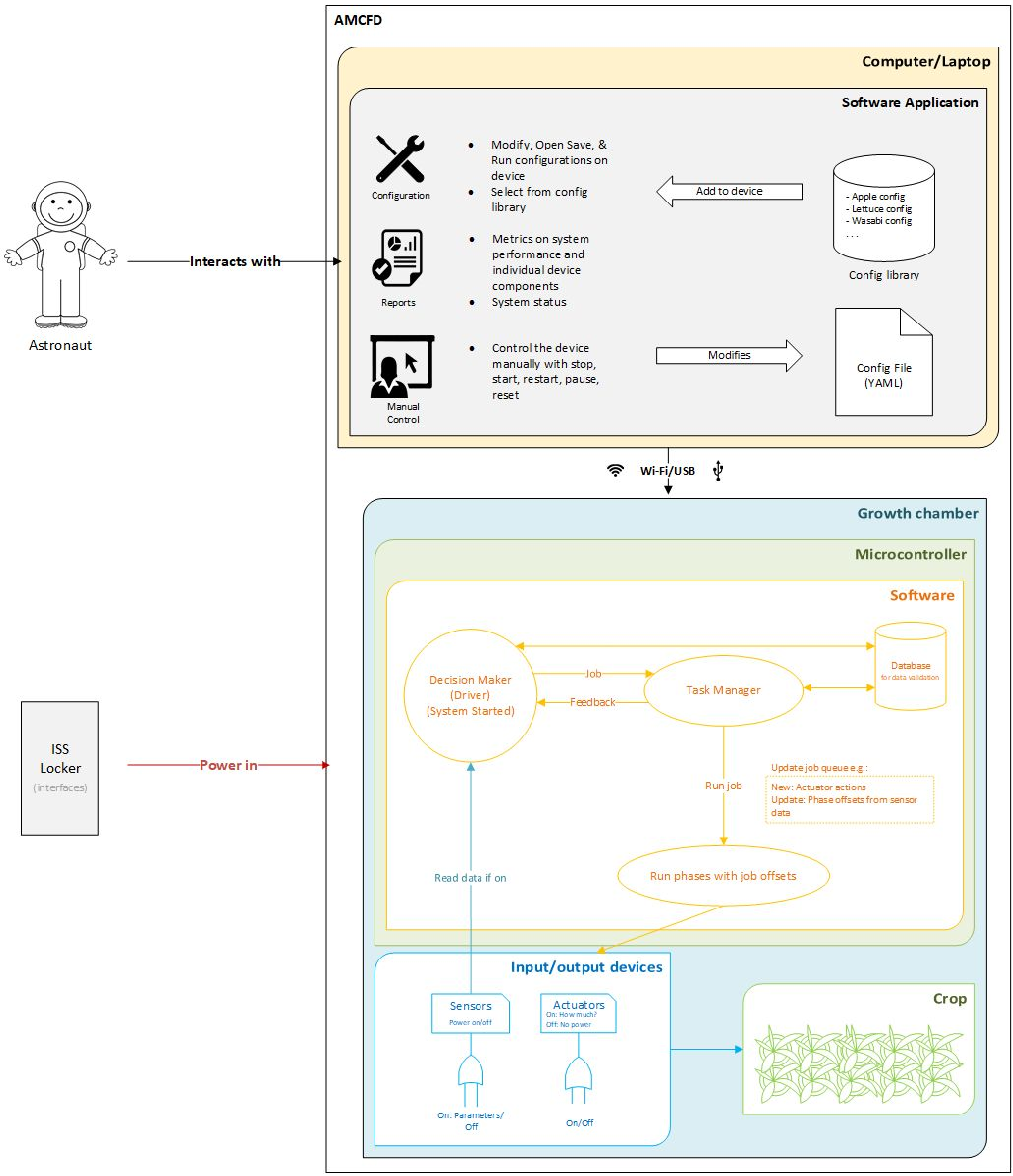
Action	Software Abstraction
Remove dirt and other particulates	Control actuators to move unused particulates to a safe disposable location for recycling
Harvest & prepare for the next cycle	Reorder all actuators and queues for the next cycle

[3] Using AI to analyze visual data is an enormous project on its own and it may not be completed with the first POC.



Approach - Application

The application will be a one-stop-shop for managing and configuring the device(s). Communication to the device will be done over Wi-Fi, but for our system it will be done over USB for now. The application will give the user reports for devices globally and individually. A feature will need to be created to facilitate adding configuration files for specific devices and specific crop types. Potentially the seed configuration files can be made available by ground control through the ethernet connection so that configurations available match the seeds onsite.


AMCFD
Computer/Laptop
Software Application
Configuration
Modify, Open Save, & Run configurations on device
Select from config library
Add to device
- Apple config
- Lettuce config
- Wasabi config
. . .
Config library
Reports
Metrics on system performance and individual device components
System status
Manual Control
Control the device manually with stop, start, restart, pause, reset
Modifies
Config File (YAML)
Astronaut
Interacts with
Wi-Fi/USB
Growth chamber
Microcontroller
Software
Decision Maker (Driver) (System Started)
Job
Feedback
Task Manager
Database
for data validation
Run job
Update job queue e.g.:
New: Actuator actions
Update: Phase offsets from sensor data
Run phases with job offsets
Read data if on
ISS Locker (interfaces)
Power in
Input/output devices
Sensors
Power on/off
Actuators
On: How much?
Off: No power
On: Parameters/ Off
On/Off
Crop

Novel Features

1. Low-power autonomous multiple-cycle growth and harvesting of food for astronauts driven by software.
2. Utilizing the latest in sensor, actuator, and software technology to protect and optimize throughput of said food.
3. Detecting diseased or under-grown plants using vision-based data.
4. Web application that can run on Mac, Linux, and Windows to control autonomous devices.
5. Library to store and access configurations for autonomous devices.

Technical Challenges

- Programming microcontrollers, possibly in conjunction with and/or parallel to other single board computers.
- Scheduling of sensors and mechanical components to reduce energy consumption and maximise throughput.
- Communicating between sensors and mechanical components.
- Processing and abstracting sensor data.
- New algorithms no one on the team has used before.
- Client/Server design
- Enterprise web application
- Designing effective reports
- USB and Wi-Fi communication

Fourth Milestone - Due February 17th

- Design for application used to manage device
    - Configuration
        - Design for configuration library service where users can select new configurations for device, i.e. apples, lettuce, etc.
    - Reports
        - System performance and status globally down to each individual subsystem of a single device
        - Appropriate metrics and report hierarchy based on user selected feature
    - Manual control

        - Stop, start, restart, pause, and reset
    - UX/UI survey for CS Forum
        - Appropriate hierarchy based on what user selected
- One production actuator working on control board
    - Not working in system
- Testing progress

Task Matrix for Milestone 4 - Due February 17th

Task	Giampiero	Christopher
Design for application used to manage device: Configuration Design for configuration library service where users can select new configurations for device, i.e. apples, lettuce, etc. Reports System performance and status globally down to each individual subsystem of a single device Appropriate metrics and report hierarchy based on user selected feature Manual control Stop, start, restart, pause, and reset	50%	50%

UX/UI survey for CS Forum with appropriate hierarchy based on what user selected		
One actuator working on production control board	50%	50%
Testing progress	50%	50%

Fifth Milestone - Due March 23rd

- Functional setup of actual actuators and sensors working with system
  - Seeding head on track
  - Light and fan
  - Linear actuator
  - Nema motors
  - Moisture sensor
- Working POC of application to configure and manage device
- Working config download service - low priority should be easy
- Metrics for reports
- Testing rules in SonarQube
- Testing scripts

Sixth Milestone - Due April 20th

- Finalized build of device with system
- Finished application
- Finished download service
- Final user manual
- Finished reports
- Testing
- User manual complete draft
- Evaluation on:
  - Crop yield
  - Multicycle experiments and tests
  - Failure and success criteria

Signature of CSE Students:

Signature: ___________________________________ Date: __________________

Signature: ___________________________________ Date: __________________

Approval from Faculty Sponsor:

"I have discussed with the team and approved this project plan. I will evaluate the progress and assign a grade for each of the three milestones."

Signature: ___________________________________ Date: __________________