# Creating Individual Unit Process and Component Designs with the AguaClara Design Tool

Heidi Rausch

December 14, 2012

## Abstract

The AguaClara student project team at Cornell is split up into teams focused on research and teams focused on design. The design team's work focuses improving and updating the AguaClara Automated Design Tool. The main purpose of the design tool is to allow communities or groups who are interested in AguaClara technologies to be able to request one of our water treatment plant designs online. Until recently it was only possible to request a design for an entire water treatment plant online; however, now, though the creation of component designs it will be possible for individual parts of the plant to be requested separately. Once code is written for a specific part of the plant, an online form will then be created so that that part can be ordered. I have created some basic guidelines for creating upper level code for individual plant components in this Google Doc. These guidelines will be revised as we develop more experience creating separate upper level design files. All of the component design files can be found in the Automated Design Tool (ADT) designs file, which can be found in the Final Designs folder on the AguaClara repository on SourceForge. In the ADT designs folder there is a separate folder for each design component that can be (or will be able to be) ordered online by an interested party. In each component's folder there should be a Mathcad file from which the online form will pull information to create the requested design, a method (and corresponding beta method) file which communicates between the user inputs, the drawing code and the LabVIEW code that outputs information to the user online, and an About file which can be used to see what variables are undefined within your code (this is useful for troubleshooting your Mathcad code). The method file is extremely important to make the online interface work correctly with the drawing code; I have created a method template so that in the future any design team member will know how to create a method file. The method template file is written in Excel, and can be found in the ADT designs file; all actual method files should be saved as tab delimited text files, should all be titled Method.txt and saved within their modular design's folder in ADT designs. The modular designs that have been created are the LFOM (Linear Flow Orifice Meter), the sedimentation tank, the

flocculation tank and the entrance tank. In the future hopefully modular designs for all individual components of the plant can be created.

# Introduction

The AguaClara project was founded in 2005, with work on an automated design tool starting soon after that. The purpose of creating a design tool was so that implementation partners, individual communities, or other interested parties anywhere in the world could request a design for an AguaClara water treatment plant. At the top level two applications, a web service and a design engine, work together to create a dynamic website that posts stock designs and that creates new web pages for new designs as they are requested. The web service, coded using LabVIEW, processes web requests for designs and checks in the bank of designs to see if an identical design has already been created. If the design has been created the client is immediately directed to the webpage of that design. If the requested design doesn't exist the request is saved to a folder for incoming requests and the Design Engine then processes that design request. The Design Engine is controlled by method files. The method files are text files which tell the design engine which version of the Mathcad code, created by students on the AguaClara design team, to use to create any incoming design requests. There are a few different types of method files. First there is one method file that controls which version of every upper level file should be used for creation of the requested design. Upper level file refers to the Mathcad file that design tool uses to create each component design; there is an upper level file for the entire water treatment plant, as well as for all the component designs that have been created and are discussed below. In the method file different version numbers can be used for different components, so if for example the sedimentation tank code was recently updated with version 5550, but that version of the flocculator code is currently being edited, the flocculator's version number can stay at 5549 or lower. The other type of method files are method files that are associated with each component design. Each upper level design file is contained within its own folder where a method file also exists. This method file serves as the basis for the online form from which design requests are made and also creates the stock designs. In this method file there is a list of all the user inputs, their descriptions, their minimum and maximum values and a default value for each. There is also a list of each stock design that we want to provide online to any interested parties who just want to quickly be able to see what a 30L/s sedimentation tank would look like, for example. Finally, there is a list of all the AutoCAD (AC, AI) variables that create the text scripts which create the AutoCAD drawings and online previews of the component design. The structure of these method files can be seen below in table 1(the italicized words are the same for each file, but anything else changes from component to component). Stock designs use the default user inputs based on the design of the plant. On the user interface online the links to each stock design's page can be found right next to the form where the user can request a more specific design if they wish

to. In addition to each of these method files there is a beta method file that parallels each one. The beta method files are exactly the same regular method files except they control the beta server instead of the main design server. The use of AguaClara's beta server is described below. The new design server website for requests is thispageand the new beta server website is this page.

Once the design engine sees that a specific design request has been made or that a version number has changed (thus triggering the production of stock designs), it immediately starts working to create that design(s). The Design Engine uses Mathcad, AutoCAD and Microsoft Word as helper applications. The Mathcad files used are used to calculate all the information that goes into the design and drawing of the component, AutoCAD is used to create the drawing of the component itself, along with any pictures that are produced for that design, and Microsoft Word creates a design specification document, both in Spanish and English (this is currently only available for plant designs). The first step in creating the design for a plant is to acquire the desired plant flow rate from the user; this is done through the online user interface (plant flow rate is the only user input required when ordering a plant). Once a plant flow rate is known, the Design Engine inserts the flow rate into the top level Mathcad file, then tells Mathcad to open the file, recalculate, and save the file, and then the Design Engine reads the saved file to extract the calculated variables. The Design Engine sends the AutoCAD scripts that were read from the Mathcad file to the AutoCAD command line using the .NET framework. The completed drawing file, selected images of the drawing, customized reports, as well as specification files are attached to a web page and the link to that page is sent to the user via email. Another option for someone wanting to request a plant would be to look at our stock designs (also available at both our design server and beta server web pages). For the stock designs AutoCAD drawing files and design specification files are available, as well as online previews of design.

In 2009 AguaClara's online software was used to design a plant for the first time. The design tool has been changed and updated many times since then and now very accurately represents the plants that our program has designed and that have been built in Honduras. The design tool is evolving rapidly, changing each semester based on new research completed by student research teams, or suggestions made by the AguaClara program of Agua Para El Pueblo (one of our implementation partners) in Honduras. Recently it was decided that individual component designs should be available because some communities could benefit from a single part of an AguaClara plant instead of needing an entire water treatment plant. Component designs could also be useful for communities who need to replace an outdated or older part of their existing plant. Designs for individual unit processes that will be integrated into an existing water treatment plant will have more design constraints and thus there will be a larger number of user inputs for the component designs than there are for the design of an entire water treatment plant.

| Description of the component design | Title | Sedimentation tank | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| *Input Parameters* | | | | | | | *Stock Designs (filenames)* | | |
| *Variable Description* | *name* | *units* | *min* | *max* | *default* | *increment* | 3Lps | 50Lps | 100Lps |
| flow rate | Q.Plant | L/s | 3 | 200 | 10 | 1 | 3 | 50 | 100 |
| *AutoCAD drawings* | | | | | | | | | |
| *Filename* | *AC Variables* | | | | | | | | |
| Sed Tank | AC. SedTank | AI. SedTank | | | | | | | |

Table 1: Method File Structure

# Beta Server Use

The beta server is currently set up as a continuously updating server (version number constantly updating to the most recent version for all modular designs we currently want to output online), so anytime a design team member commits a file to our SourceForge repositories, the beta server will restart making all the stock designs with that revision number. The continual updates allows anyone who commits changes to see, in a few minutes (or possibly up to an hour depending on what they changed), how their change affected the plant design by checking online on the beta server website. The version number that the design server is using for an specific piece of the plant is controlled by the overall method file in the ADT designs file. When someone is interested in whether or not the latest changes they committed are working correctly, they can check the about file, which is also displayed on the beta server website, for the plant or for the individual modular code piece they were working on. The about file lists all the important variables for each piece and if they were not calculated an error message is shown, so it is easy to identify mistakes. About files are created each time any component is designed by the design server. The

variables they contain are based on a list of "display variables" found in each upper level design file. The display variables are simply listed in each upper level file as undefined variable statements which are then filled in and shown in the about file as they are produced.

## Changes to the Master Water Treatment Plant Design file

The main upper level file, which is used to create AguaClara plant designs and drawings is called EtFlocSedFi (Entrance Tank, Flocculator, Sedimentation Tank, Filter). This is the file that the Design Engine originally calls upon when it receives a request for a plant design, or when the version number on either the design or beta server is changed. EtFlocSedFi has references to every design and drawing file necessary to create a design of the plant. It also includes definition statements of all important variables related to the plant so that they will show up in the about file produced by the design server.

When beginning to develop the modular designs it was determined that many individual plant component design and drawing files called upon variables only available in another specific plant component file. For example, the flocculator design code calls upon variable created and stored in the sedimentation tank code. This type of coding is fine when drawing the entire plant, as each file is used to help draw the plant, so all variable are created and can be called upon in later pieces of the plant drawing. However, when trying to create a design of a stand-alone flocculator, how should the variable normally calculated within the sedimentation tank design file be found? The best option is to remove variables that depend on previous code from their individual files and place them above their file references in the EtFlocSedFi file. By doing this, these variables that are carried over from one plant component to the next can become user inputs (in most cases). With regards to the drawing files, at least the origin points of each individual plant component (except the sedimentation tank) will have to be pulled from the drawing file and placed in EtFlocSedFi. This is due to the fact that each part of the plant's origin point is based on a previous plant component's origin point; by moving each component's origin point to EtFlocSedFi the origin points for each modular design can be redefined as (0,0,0). This seems like it may clutter up EtFlocSedFi but it will also easily show how each part of the code depends on the code above it. In figure 1 below you can see how EtFlocSedFi with the new changes.
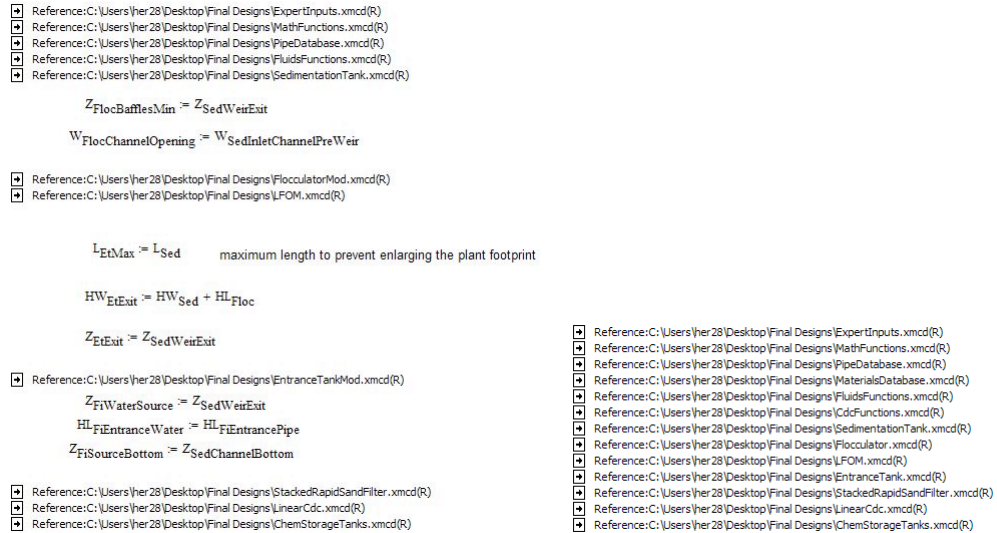
Reference:C:\Users\her28\Desktop\Final Designs\ExpertInputs.xmcd(R)
Reference:C:\Users\her28\Desktop\Final Designs\MathFunctions.xmcd(R)
Reference:C:\Users\her28\Desktop\Final Designs\PipeDatabase.xmcd(R)
Reference:C:\Users\her28\Desktop\Final Designs\FluidsFunctions.xmcd(R)
Reference:C:\Users\her28\Desktop\Final Designs\SedimentationTank.xmcd(R)

$$Z_{FlocBafflesMin} := Z_{SedWeirExit}$$

$$W_{FlocChannelOpening} := W_{SedInletChannelPreWeir}$$

Reference:C:\Users\her28\Desktop\Final Designs\FlocculatorMod.xmcd(R)
Reference:C:\Users\her28\Desktop\Final Designs\LFOM.xmcd(R)

$$L_{EtMax} := L_{Sed} \qquad \text{maximum length to prevent enlarging the plant footprint}$$

$$HW_{EtExit} := HW_{Sed} + HL_{Floc}$$

$$Z_{EtExit} := Z_{SedWeirExit}$$

Reference:C:\Users\her28\Desktop\Final Designs\EntranceTankMod.xmcd(R)

$$Z_{FiWaterSource} := Z_{SedWeirExit}$$
$$HL_{FiEntranceWater} := HL_{FiEntrancePipe}$$
$$Z_{FiSourceBottom} := Z_{SedChannelBottom}$$

Reference:C:\Users\her28\Desktop\Final Designs\StackedRapidSandFilter.xmcd(R)
Reference:C:\Users\her28\Desktop\Final Designs\LinearCdc.xmcd(R)
Reference:C:\Users\her28\Desktop\Final Designs\ChemStorageTanks.xmcd(R)

Reference:C:\Users\her28\Desktop\Final Designs\ExpertInputs.xmcd(R)
Reference:C:\Users\her28\Desktop\Final Designs\MathFunctions.xmcd(R)
Reference:C:\Users\her28\Desktop\Final Designs\PipeDatabase.xmcd(R)
Reference:C:\Users\her28\Desktop\Final Designs\MaterialsDatabase.xmcd(R)
Reference:C:\Users\her28\Desktop\Final Designs\FluidsFunctions.xmcd(R)
Reference:C:\Users\her28\Desktop\Final Designs\CdcFunctions.xmcd(R)
Reference:C:\Users\her28\Desktop\Final Designs\SedimentationTank.xmcd(R)
Reference:C:\Users\her28\Desktop\Final Designs\Flocculator.xmcd(R)
Reference:C:\Users\her28\Desktop\Final Designs\LFOM.xmcd(R)
Reference:C:\Users\her28\Desktop\Final Designs\EntranceTank.xmcd(R)
Reference:C:\Users\her28\Desktop\Final Designs\StackedRapidSandFilter.xmcd(R)
Reference:C:\Users\her28\Desktop\Final Designs\LinearCdc.xmcd(R)
Reference:C:\Users\her28\Desktop\Final Designs\ChemStorageTanks.xmcd(R)

*Figure 1: New and Old versions of EtFlocSedFi*

# LFOM (Linear Flow Orifice Meter)

The LFOM was one of the higher priority modular designs, so it was the first one to be worked on. The first thing to be decided when creating a modular design was exactly what we wanted the code to output. For the LFOM it was decided that all that was needed to be drawn was simply the pipe with the orifice holes in it. Other parts of the rapid mix pipe and entrance tank do not need to be included because we do not know the context in which the LFOM will be used by whoever requests it. User inputs also need to be chosen for each modular design. Every piece must have Q.Plant (the flow rate of the plant (Q.Plant) as an input, but most parts will need to have other inputs as well. Most pieces will need other user inputs because, now that the piece is not incorporated into the plant, additional aspects of the design that previously relied on other plant components will have to be supplied by the user. For the LFOM, there are simply two inputs, Q.Plant and HL.Lfom, which is the head loss across the LFOM. After the user inputs are stated in the code, then references to expert inputs are made. Expert Inputs is a Mathcad design file that specifies constants in the AguaClara plant design. The entire expert inputs file is not referenced as not all the information in it is necessary for drawing an LFOM, or any other part of the plant for that matter. After the expert inputs references, there is a section that includes references to other important files in the AguaClara Final Designs folder, which are MathFunctions, FluidsFunctions, MtoATranslators, MtoATranslatorsWrapped, PipeDatabase and PlumbingF. These must be referenced in all modular design files, each of these files contain basic code that has bee established by previous AguaClara design teams and is essential to creating

our plant designs. Finally, the LFOM design file and LFOM drawing code are referenced. At the end of the code, after already defining the AC (AutoCAD) variable (which lists all the commands that together will create an LFOM in AutoCAD), one of the most important things to include is an undefined statement of the AC variable. This is necessary so that LabVIEW can correctly find the AC variable and enter it into the AutoCAD command line.

At this point the folder for the LFOM in ADT designs is complete. There is the drawing code, the drawing template code (created by design team member Marlana Hinkley in the summer of 2012), the method file, and the beta method file. LFOMs can now be ordered online from the design server or the beta server. An example of a LFOM with HL.Lfom set to 20 cm and Q.Plant set to 20L/s can be seen in 2 below.
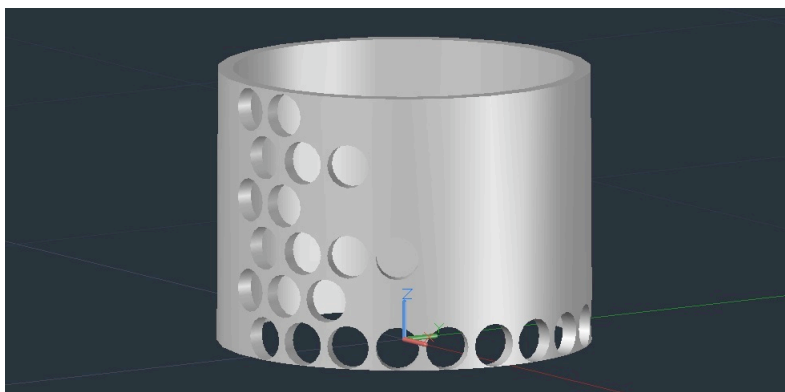


Figure 2: LFOM Example

Please note that the LFOM modular design was the first to be developed and therefore I have included specific details about the process I used when creating this design. For all the following designs I used a similar process, but did not include as much detail about some of the more mundane steps, instead I simply discussed the difference or difficulties that came up each in case.

## LCDC (Linear Chemical Dose Controller)

The second modular design that was worked on a design for the chemical dose controller. The chemical dose controller is one of the most important parts of the plant and is one of AguaClara's most innovative concepts; having a modular design for this plant component could prove extremely helpful since the LCDC could be used to dose chemicals in many different contexts. However, the LCDC proved to be significantly harder to break away from other parts of the original plant code than the LFOM. Difficulties were created by the fact that many of the doser's components' elevations were defined by other parts of the plant. Other problems came up when trying to design a single doser that could work

for any of the three chemicals we use in our plants. It seems most logical for a modular design of the doser to be a single doser, however then the question of what chemical to dose is a significant issue. Much of the design of each doser depends on the viscosity of each chemical. Therefore, to make the code as general as possible many different options for chemicals would have to be coded in to be chosen by the user. It has not yet been decided which and how many chemicals the code should be able to work for. For this reason work on the LCDC modular code has been put on hold.

## CDC Plant Drawing Code

Work on the CDC drawing code for the plant also needs to be done in order to finalize the design within the total plant design and the LCDC modular design. Recently sliders have been added to the plant drawing code for the CDC system, they can be seen in figure 3 below. Other recent corrections to the drawing include accurate placement of the fittings for the connector tube to attach to and correct subtractions of many of the pieces attached to the lever arm. Finally, the option of one, two or three dosers has also been added. Three doser can be seen in figure 4, two dosers (which would be chlorine and plant coagulant) can be seen in figure 5, and one doser (currently the chlorine doser) is seen in figure 6.
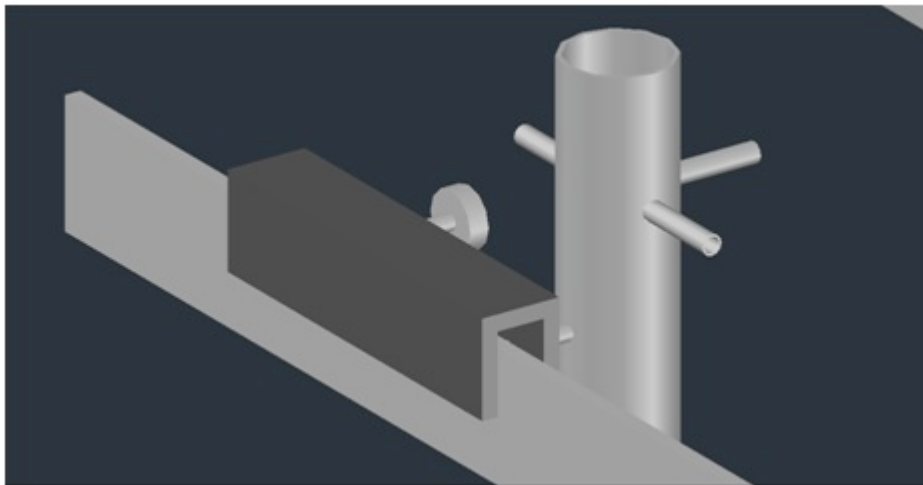


Figure 3: Slider and Fittings
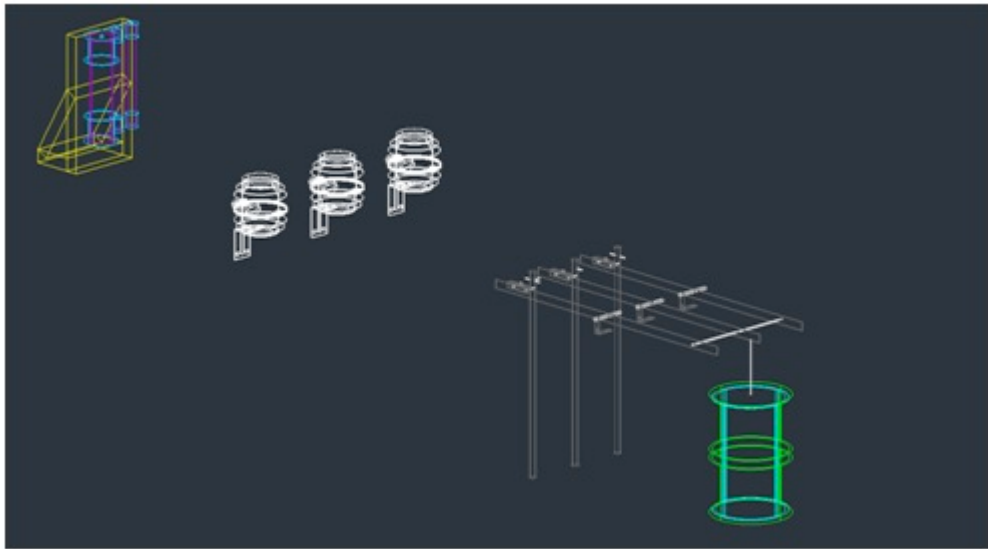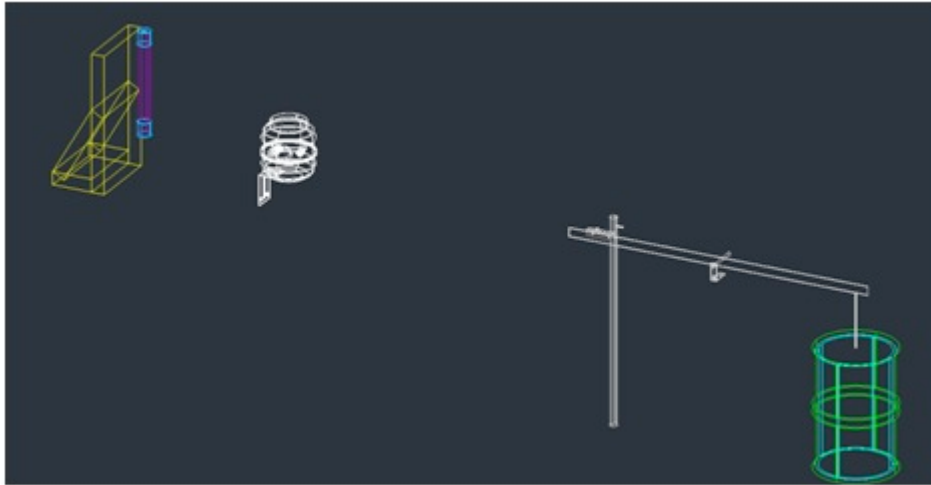
Figure 4: Three Dosers
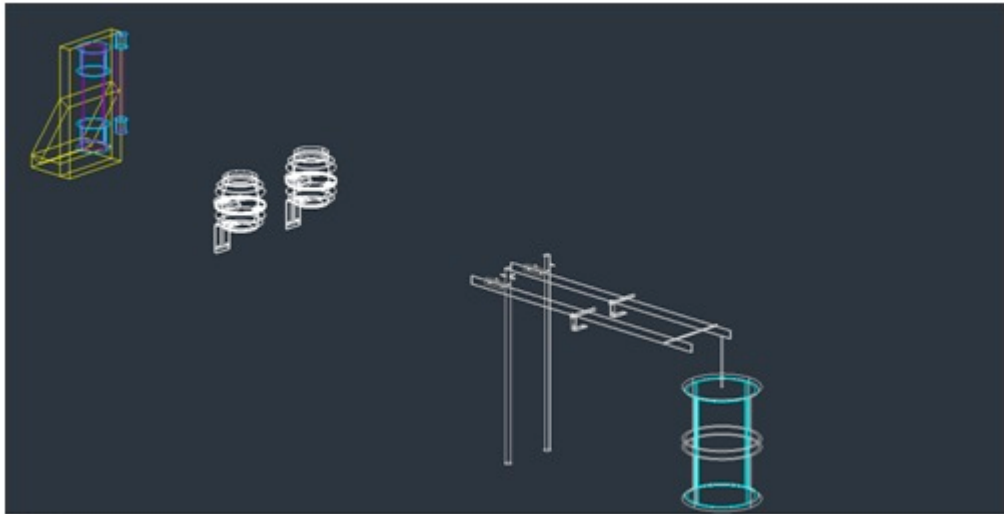


Figure 5: Two Dosers

Figure 6: One Doser

.

The new CDC manifold system (developed in the summer of 2012) and the CDC lever arm system (developed in the fall of 2012) needs to be added to the design and drawing. The recent changes by the Chemical Dose Controller research team will be incorporated into the design tool in the spring of 2013.

## Stacked Rapid Sand Filter (SRSF)

The SRSF was invented by the AguaClara team in 2010 -2011 . The first full scale AguaClara SRSF went online in October of 2011. Using the same method as had been used to develop LFOM modular code a filter modular code was also developed. However, similarly to the LCDC design it quickly became evident that the design and drawing of the filter relied strongly upon the design of the plant itself. Specifically, many of the elevations of various parts of the filter were highly dependent on elevations of other parts of the plant. The beginnings of a modular filter code have been started but some of the elevations are clearly very wrong, as can be seen in figure 7.

There are other still unanswered questions about how to most appropriately design a stand-alone filter. For example, what type of inlet and outlet system should be included with the design for the filter; this could be different for every community based on how they currently receive and distribute their water source. These types of changes will be hard to make while still keeping the plant filter code and modular filter code one and the same.
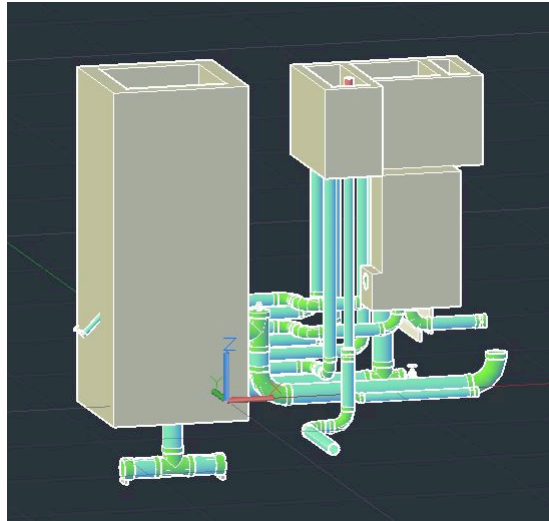
Figure 7: Filter with Inaccurate Elevations

# Sedimentation Tank

The modular code for the sedimentation tank was the easiest component to design. The sedimentation tank is one of the more basic features of the plant and was also the first to be coded into the AguaClara design tool. Therefore, the origin point of the sedimentation tank is the origin point of the entire plant and the other plant components were written to be dependent on the sedimentation tank design. Currently the only input for this modular design is a flow rate through the tank(s). Below you can see a 20L/s design in figure 8. In the future the user inputs will be expanded for this design, giving the requester more control over how they want their sedimentation tank to be constructed. User inputs that will be added include plate settler width, capture velocity, energy dissipation rate and maximum inlet manifold diameter. The additional of these user inputs will give the user more flexibility in designing a sedimentation tank that will for best for their specific situation.
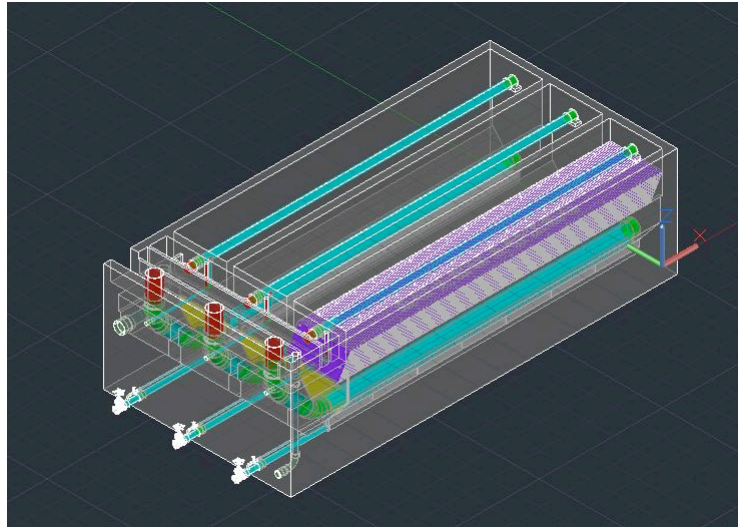
Figure 8: Sedimentation Tank Modular Design

# Flocculator

The flocculator required more work than the sedimentation tank, however it was not too difficult to complete. The flocculator requires extra user inputs as the flocculator design in the plant is dependent on the sedimentation tank design. The current user inputs for the flocculator are not ideal and are not necessarily parts of the flocculator that the user would have knowledge of, or be interested in changing. The user inputs that will be used in the final version of the code are the energy dissipation rate, the collision potential, the depth of water at the exit of the flocculator and the length of one channel of the flocculator. Below in figure 9 a flocculator 20L/s design with user inputs that would match a 20L/s plant can be seen.
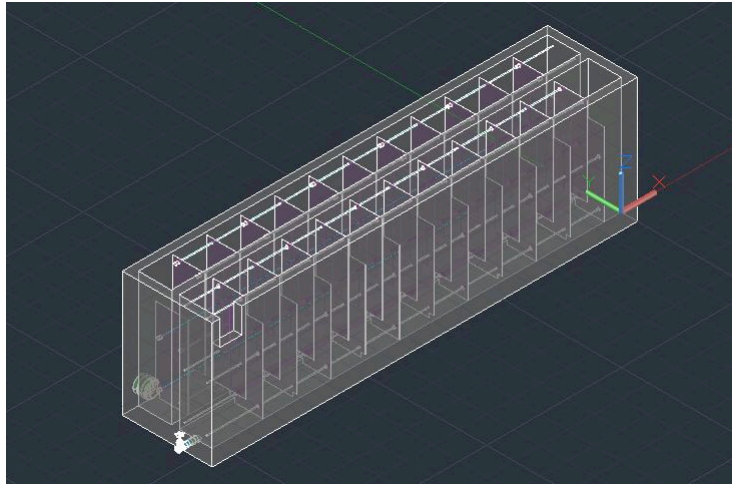
Figure 9: Modular Flocculator design

## Entrance Tank

To create a entrance tank modular design, several user inputs were again required. Also to complete this design several elevation definitions were removed from the entrance design and instead these elevations are simply defined as a point plus the height required to reach that elevation. This allows these heights to be redefined for the modular code. Ultimately it was determined that the user needs to input a flow rate, a maximum length of their entrance tank, the elevation of water exiting the entrance tank and the elevation of the pipe which allows water to exit the entrance tank. Below in figure 10 an entrance tank 20 L/s design with user inputs that would match a 20 L/s plant can be seen.
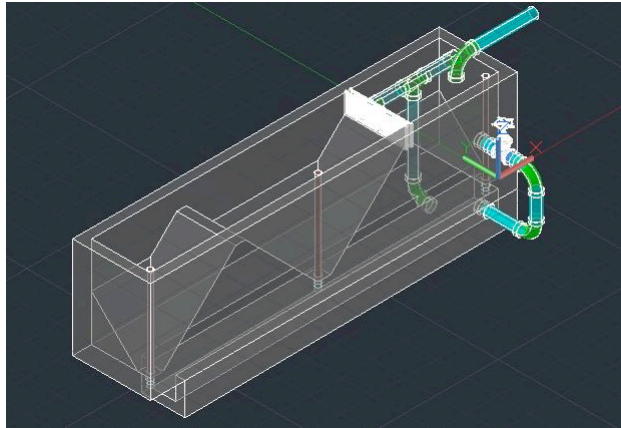
Figure 10: Entrance Modular Design

## Turbidimeter

In addition to creating modular designs for plant components it was also suggested that a design be created for a turbidimeter. The turbidimeter described here is a handheld, simple, inexpensive device created by AguaClara students to measure the turbidity of water when conventional turbidimeters are not available. Below in figure 11 a preliminary drawing of a turbidimeter can be seen. Some future improvements to this drawing could be making the light in the middle look more realistic, adding the design on the top of the viewing disk as well as adding lines to show turbidities.
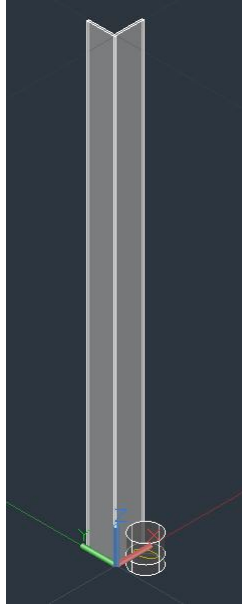
Figure 11: Turbidimeter AutoCAD drawing

# Low Flow Plant

Recently work was begun on a new plant layout for low flow plants. Low flow plants are AguaClara plants that are designed to serve communities who have flow rates of 6 L/s or less. Another student on the AguaClara design team, Julia Morris, created design modifications for almost all plant components and has pieced them together to create a new low flow plant design. Soon users will be able to obtain plant designs for flow rates between 0.8 and 5 L/s online. Below in figure 12the previews of a complete plant designed for 0.8 L/s and a sedimentation tank and flocculator can be seen.
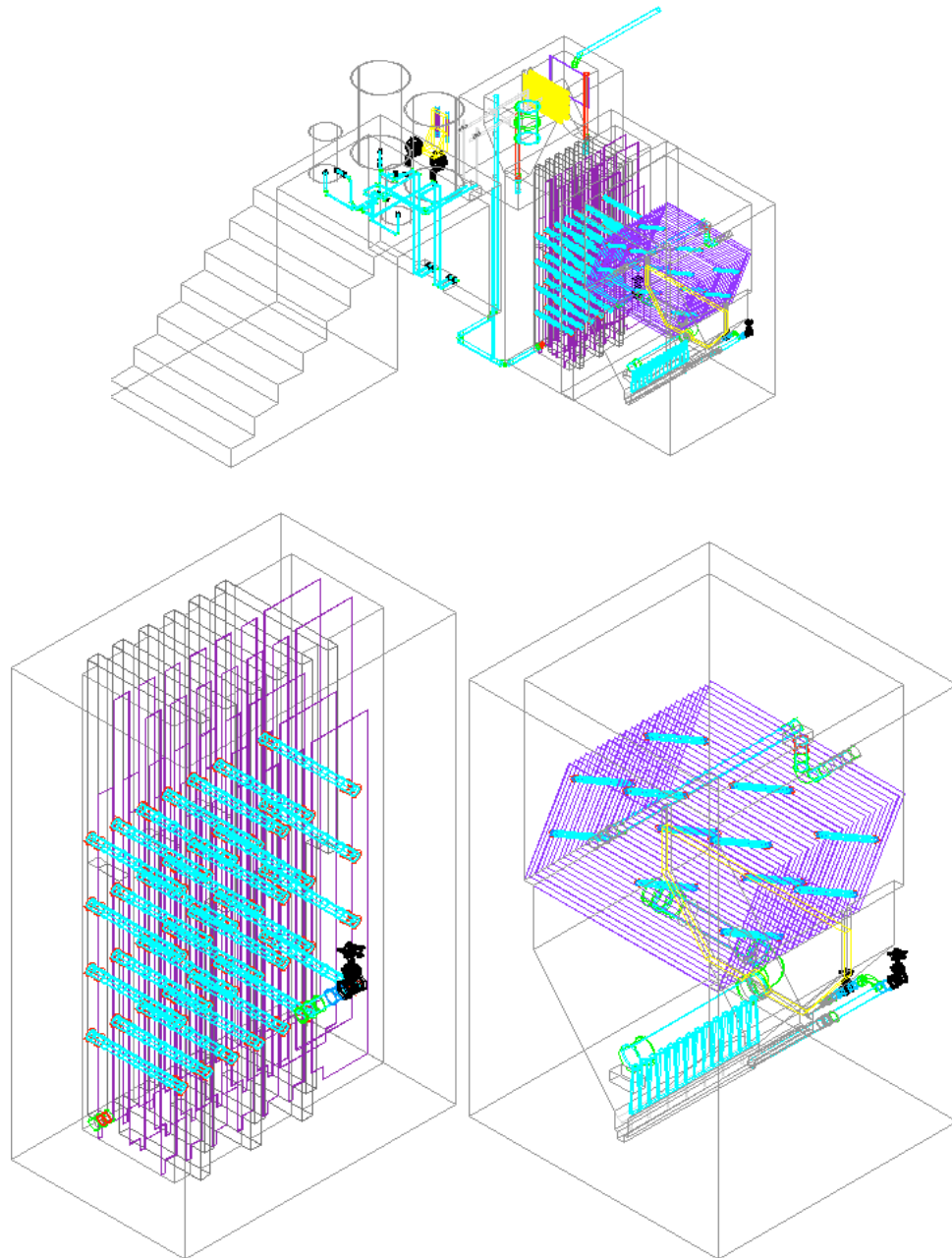
Figure 12: Low Flow Plant, Sedimentation Tank and Flocculator

# Conclusions

The AguaClara automated design tool has changed and improved significantly since it's beginning. The addition of individual component designs for each part of the plant will hopefully only improve it further and will prove to be of great use in the future. The problems that came up when creating these component designs do suggest some guidelines for future work on updating the design tool. For example, the problems that were created by the fact that all of the plant component's original files were so interconnected could have been avoided if the individual files had been written to be more modular from the beginning. In the future when updating different parts of the plant codes the design team can make an effort to not introduce more of these interdependencies. Another lesson that I learned from working on this project was that there could be stricter guidelines about how code within a team should be written. Though most of the files within the AguaClara repositories are written similarly, they are not all identical in structure.