The following document will walk you through how the code of the Michigan Legislative Modeling Tool works, created in 2024 by Master's of Engineering Applied Climate student Brian Cromwell at the University of Michigan. Tool created in partnership with the Michigan Department of Energy, Great Lakes, and the Environment.

Throughout this document, cell numbers will be referred to. These numbers are the values seen after restarting the code and running all cells, and line up regardless of which software you use to run the code. These numbers do not include text cells.

This guide is mainly meant to be used by anyone who wants to or needs to adjust the code. If you are simply using this tool to generate an output, this document will likely not be relevant for your needs.

Updating the Code for the Future

To preface, this code was created in 2024 using 2024 data. However, the model should be somewhat future-proofed, in the sense that it can be updated each year with new data. Updating is very easy, and only requires changes to two lines of code, both of which are in cell 2. The first is the variable currentYear, which should be set to the year for which you have the most recent data. The second is the variable CO2EmissionsHardCoded2 (See below for the reasoning behind this name). For updating this variable, add another value for the most recent data for carbon emissions across the State of Michigan in Million Megatons of Carbon Equivalent (MMT CO2e).

The Data

Cell 2 sets some initial variables, as well as creates a Business As Usual case. There are few different data sets for tracking the measured carbon output data, but the main method uses the variable CO2EmissionsHardCoded2. There is also a variable named CO2EmissionsHardCoded1, which was found in the spreadsheet but is inconsistent with some of the other reduction measures. The variable was included in the final product in case any later users want to use it.

Cells 3 and 4 pull in data for Reduction Measure 1. There are emissions data and emissions from electricity data. The electricity data is what was used in the final product, as it produces a more realistic result, but the other data is also valid and may be switched out at will should the used desire.

Cells 5-13 pull in data from Reduction Measures 2-10 respectively. All of the data for this model was taken in from the Reduction Measure Quantification Spreadsheet.

Pre-Existing Reduction Measure Tool - All or Nothing

Cell 14 is where the input parameters go. Nothing special, just a True/False toggle for each variable. This could likely be made even simpler in the future if a version wanted to have both states present and have the user toggle between the two of them, but this method works for now.

Cell 15 contains most of the code for the tool. First, the working list is fed the values of the measured data beginning in 2020. Then working through each year, an initial yearly change is set based on the business as usual case. From there, the code checks to see which reduction measures are active, adding on the yearly change to that reduction measure if it is. Once all 10 reduction measures are checked, a final yearly emissions value is calculated based on the previous year's value and the calculated yearly change. This new yearly value is added to the working list, and the process repeats until 2050. The results are then graphed against the business as usual case.

Pre-Existing Reduction Measure Tool - Split Goals

Cell 16 is an input for the split reduction measure goals, again with the variables togglable on and off using a True/False state. Cell 17 works almost identically to Cell 15 with the exception of splitting the reduction goals into 2030 and 2050 milestones. This was accomplished by checking the year and calculating the change from there. The one thing to note is that for the 2050 goals, the code checks to make sure that both the 2030 and 2050 goals are turned on, as it would be much less realistic for the 2050 goal to be hit after not reaching the 2030 goal.

Custom Input

Cells 18-27 are more data assimilation. They create a number of target percentages for each variable and interpolate between them to create a list of the percentages that each reduction measure will achieve each year.

Cell 28 creates the log and a function that will append any input reduction measures into the log. Very simple function that is used in the next function.

Cell 29 creates a function that calculates the total carbon reduction of any single custom reduction measure. It first figures out the number of the reduction measure, then pulls in that reduction measure's information. From there, there are three different cases. The code can either scale down the lowest value if the percentage is smaller than the initial value, or scale up the total value if the percentage exceeds the 2050 value, both of which are very simple. If the reduction measure percent is within the bounds of the reduction measure, it finds the year in which the percentage is closest, and interpolates a total carbon reduction value between the two nearest percentage values.

Cell 30 is where the user inputs all of the custom variables. These inputs include reduction measure number, reduction measure percentage, reduction measure year, population, and whether or not the results want to be downloaded as an output. The code then runs the customReductionMeasure function to calculate the custom carbon equivalent reduction.

Cell 31 is another input cell for the pre-existing reduction measures, similar to the previous two sections. The True and False toggle still applies here.

Cell 32 does the work of outputting the tables and graph for the custom input. First it sets up the working list, then goes to the list tracking the more specific yearly impacts for each reduction measure. The code then loops through the years from 2050. The code is very similar to the previous tools, with the addition of keeping track of the individual reduction measure impacts. The code first calculates the custom impacts from each of the measures in the log, then adds their impact to the specific reduction measure count. The code then loops through the same all or nothing process from earlier. Once the yearly changes are calculated, the population factor is accounted for before adding the final reduction measure and total impacts to their corresponding list. The tables are then set up and displayed, creating the final output.

Lastly, Cell 33 allows the user to check which measures are currently in the log and the corresponding carbon impact of these measures.