**Step 3: Plan the Solution (Design the Algorithm)**

**3.1 Algorithm (in plain English)**

Loop forever:

Start Input: Scheduled\_time, Feeding\_time, Real-time clock, Bin\_full, Bowl\_full

Check if the food bin is full. If not, turn warning light on.

If the bin is full, check if it's feeding time and the bowl is not full. If yes, the servo will turn on and fulfill the bowl with food, changing its status to bowl full.

If the bowl is still full 15 minutes after the scheduled time, turn warning light on.

**3.2 Truth Table**

|  |  |  |
| --- | --- | --- |
| Truth table | | |
| Bin\_Full | Bowl\_Full | Red\_light (Warning) |
| 1 | 1 | 1 |
| 1 | 0 | 0 |
| 0 | 1 | 1 |
| 0 | 0 | 1 |

**3.3 Boolean Expressions (SOP)**

Red\_lighon = (Bin\_full x Bowl\_full + Bin\_fullOFFx Bowl\_full + Bin\_full x Bowl\_fullOFF)

**3.4 Pseudocode**

**Loop forever**

**Start**

**Input:** Scheduled\_time, Feeding\_time, real-Time, Bin\_statu and Bowl\_status.

**If** **(**Bin\_full = 0**)** #Check if bin is full

Turn Red\_light ON

**If** **(**Bin\_full = 1 **and** Bowl = 0 **)** #Check if bin is full and bowl is empty and fulfill it

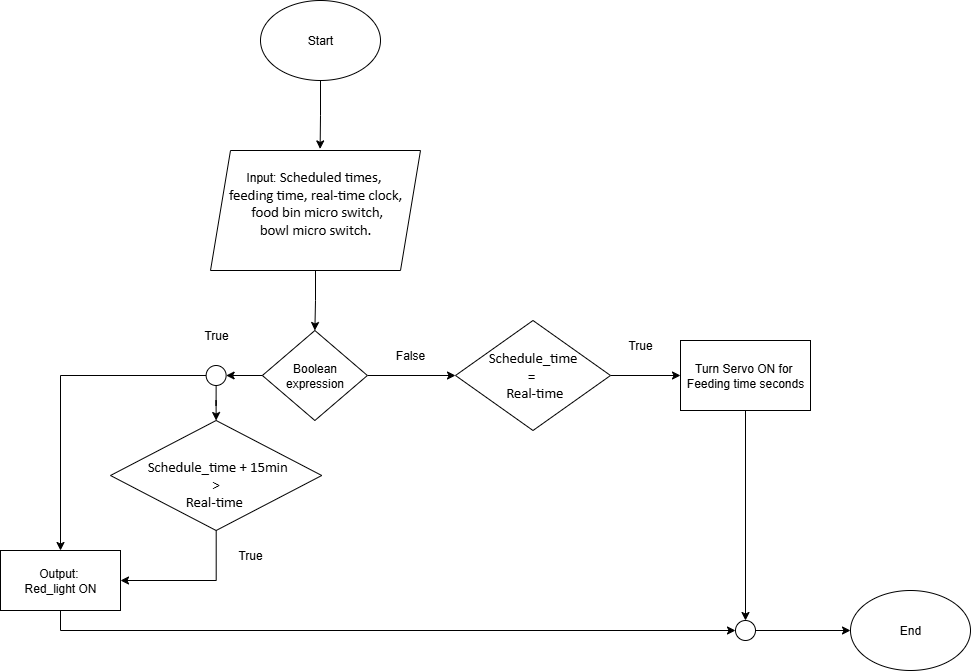
**If** **(**Schedule\_time = Real-time**)** #when it is schedule time

Turn Servo ON for Feeding\_time seconds

**If** **(**Real-time > Schedule\_time +15min **and** Bowl\_status = 1**)** #Wait 15min to do the checking if pet

Turn Red\_light ON #animal has finished the bowl

**3.4 Flowchart**



**GitHub repository link**

The link for my repository is show bellow

<https://github.com/KennySousa224/pet-feeder-project>

**References**

* draw.io. (n.d.). *diagrams.net*. Retrieved August 17, 2025, from <https://app.diagrams.net/>
* Lipovski, G. J. (1999). *Introduction to microcontrollers: Architecture, programming, and interfacing of the Motorola 6812*. Academic Press.