Main Storage Design Procedure

Disclaimer: this guide only covers the basic steps for a filter-based system by picking up IKEA modules from Storage Tech Discord and wont cover advanced topics like encoded, var sorting etc

Before we start designing a storage the first step should always be to decide the capacity, because this is the main dependency. The reason for this, is because the storage modules that we usually use are splitted in three categories:

- Bulk, a module which targets to high-capacity storage
- Chest, a module which targets to relatively small capacity
- Multi Item Sorter, a module which targets to very small capacity

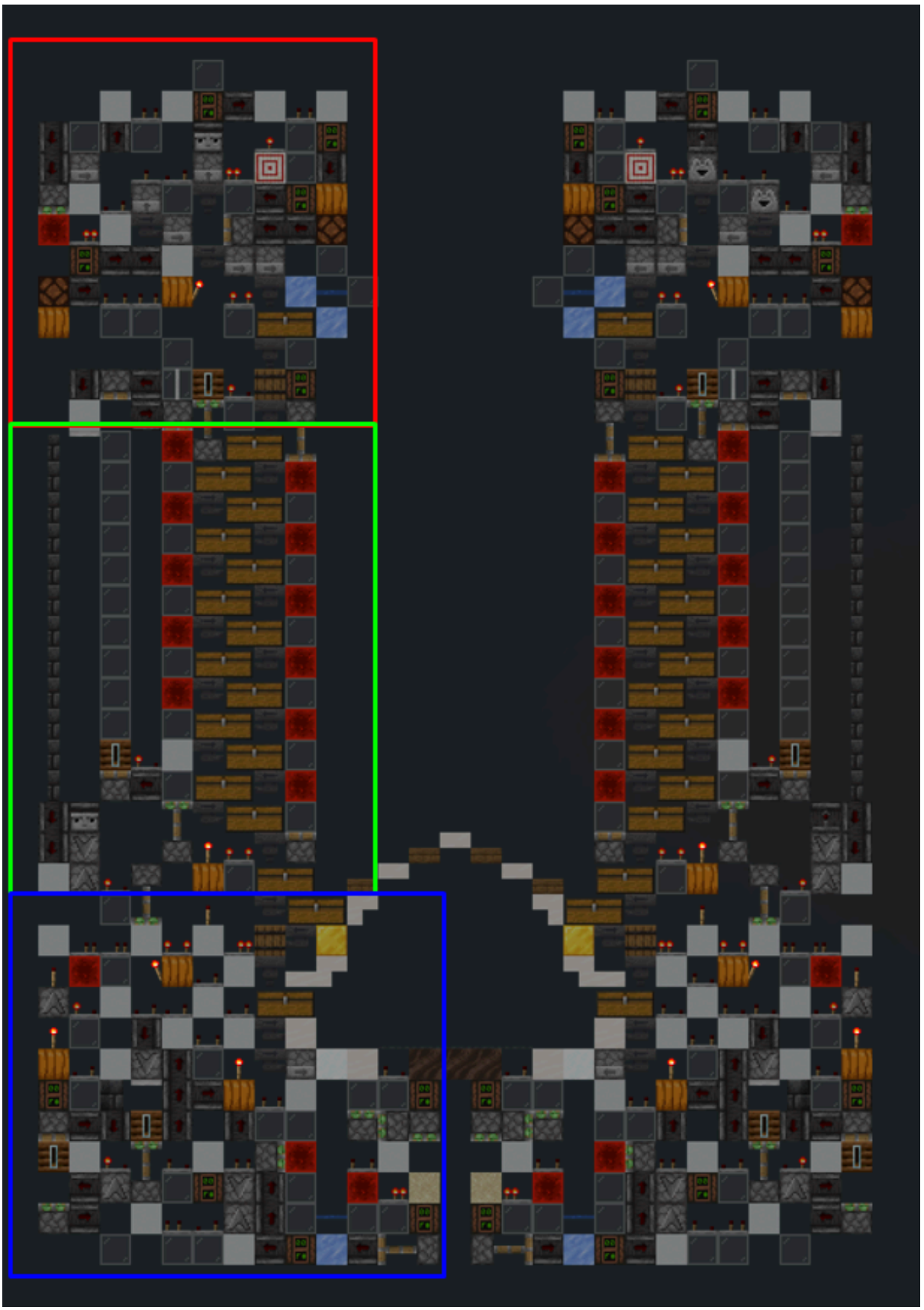
Bulk Modules:

The bulk sorters are commonly used to efficient store materials that are in high demand and easily acquired, like cobblestone, dirt, deepslate, etc.

Due to their specialised usage, in most of the cases they store items exclusively in shulker boxes, thus allowing for big quantities to be stored efficiently. To put it with raw numbers, a simple chest that contains 27 stacks of items, is capped at 1728 items. But a simple chest oriented for bulk storage, which stores shulker boxes, can store 46.656 items, thus you need much less chests to store the same amount of items, under the assumption that shulker farms (and so renewable and infinite shulker boxes) are accessible in your version.

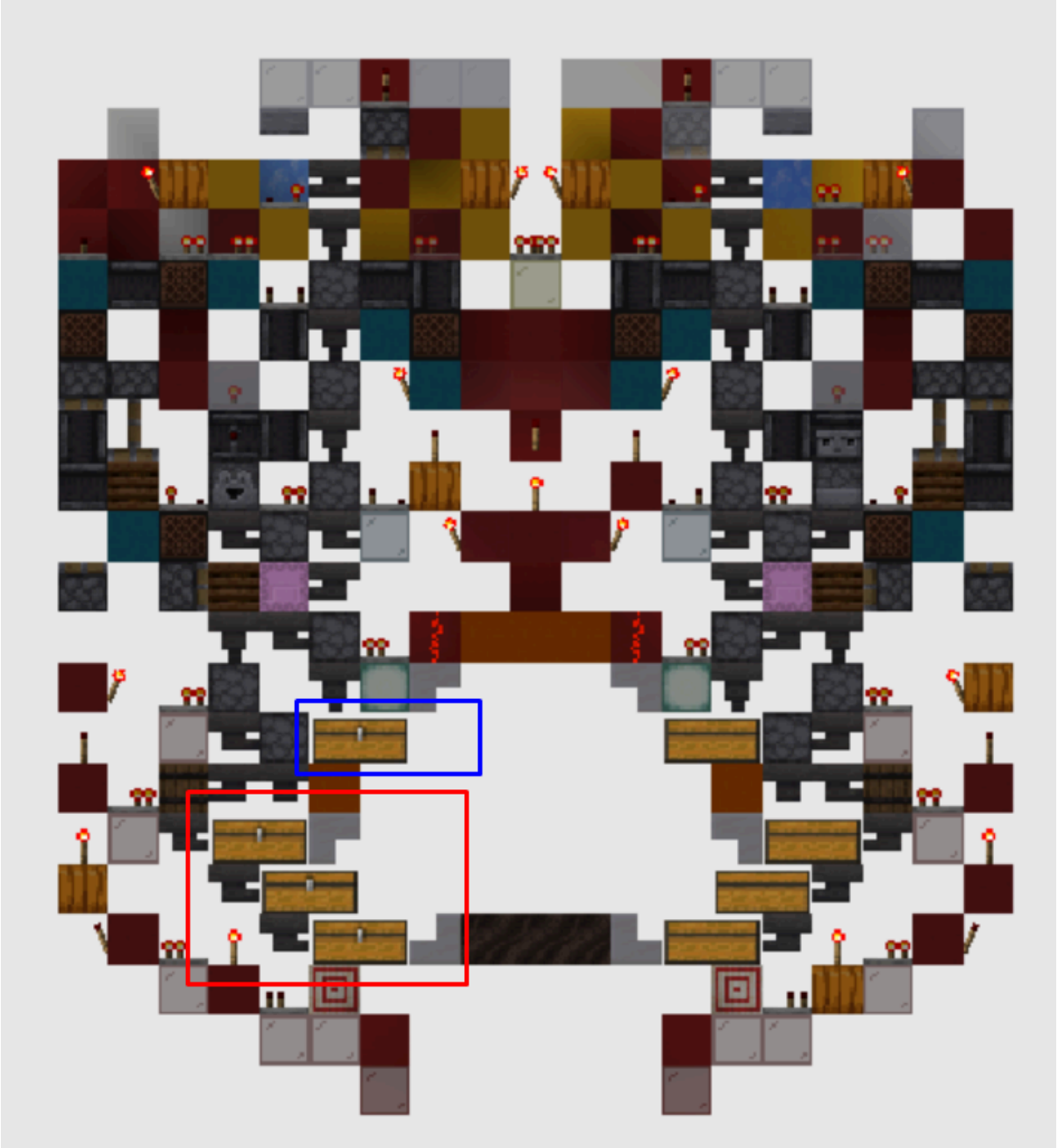
Bulk modules usually have capacity at the range of millions of items, as in the example below where we can see some things.

1. The design is symmetrical. This is very common in storage parts, because that way you can have access to more (double in this case) items without need to walk long halls.
2. Red Box: This is called **box sorter** and it's a special type of filter specifically designed to filters boxes of items without the need of unloading them.
3. Green Box: The chest tower where the boxes stored.
4. Blue Box: This part is called **Display,** and its purpose is to have the items accessible to the players. This is achieved by taking one box from the tower and placing it in the ground in a way that a player can open it and access the items. If on the other hand, you need more than 1728 items, the chest is accessible so you can grab full boxes. The gold block is indicator of which item is stored at each chest tower-slice.



1.84 Mil Smart-Restock Bulk Storage by Ners

In the above design we can store up to 1.84 million items, and for some people this might be overkill (that's why the first step is always to determine what we want). These days a new type of bulk is being developed, that is called Hybrid Bulk, which targets a smaller capacity.



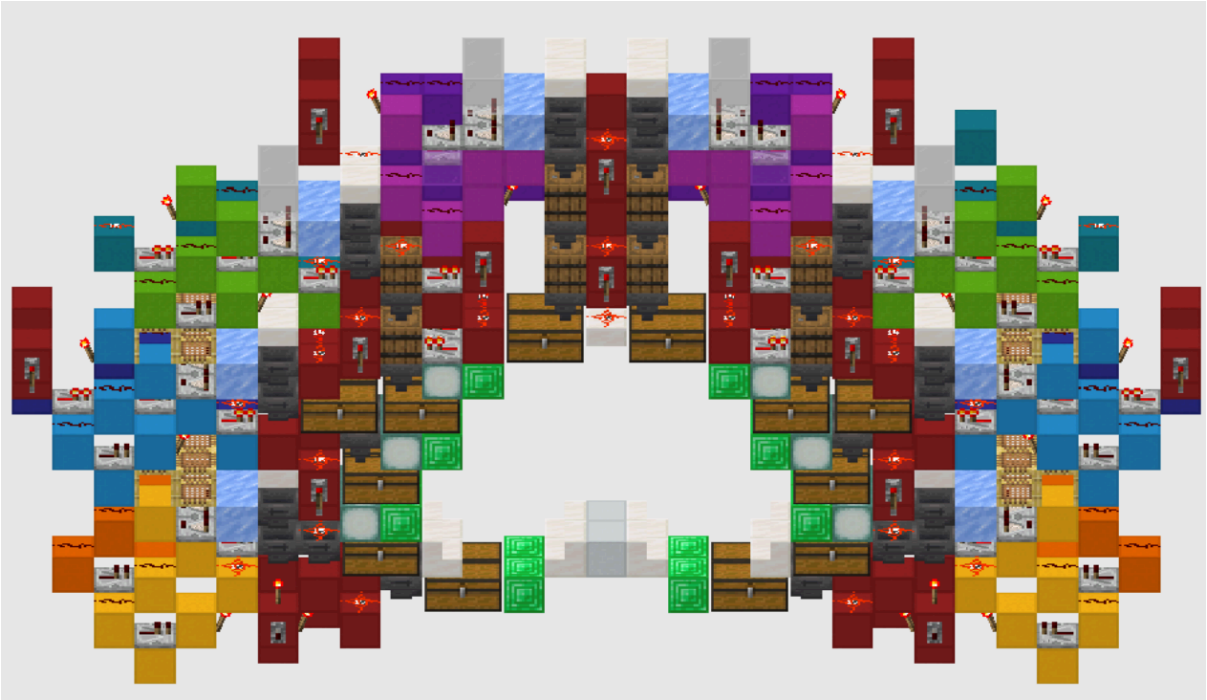
Composter based Hybrid Minibulk by restartlife

This kind of bulk doesn't use box sorters, is smaller with smaller capacity (~360k items) and the display is different. To be accurate, it doesn't use a conventional display, but it has a double chest (blue box) which contains loose items. This means that the first items you will throw, will end there. After this chest is full, then it will start filling shulker boxes, that will be stored at the bottom chests (red box).

Of course, like in the previous bulk, there is a block indicator of the item that is stored, in this case a terracotta.

Chest Halls:

Chest halls are modules that targets to smaller quantities than bulk, they don't store their items in boxes, but in chests. Usually, the designs have capacity of 1 or 2 double chests, plus some hoppers. This limits the capacity to ~3.5k items for 1 dc (double chest) and ~7k items for 2 dc.



1wABt 8 item 2dc SS2 Chest Hall By IAMBOB

As we can see in the above photo, it has 8 indicator blocks per slice, which means it can store 8 items per slice. Also, we notice the different colours in the filters. This is not a common thing, but it can help someone new to dissect a slice. Each one of the colours is a different filter, specifically made in a way so they don't interfere with neighbour filters. Also, we are noticing the levers on the red concrete. Usually this is an indicator of a **locking** mechanism. At this stage we only need to remember that these levers are only **indicators** and will be replaced when we build our storage.

The most common designs are the 8 items per slice halls, and the next are 10 items per slice, but they used less.

Multi Item Sorters

Consider cobblestone. You get a lot of cobblestone, so you could justify giving it a double chest of its own, or even store it in shulker boxes. Now think about ancient debris. *Most* players will not need shulker box storage or double chest storage for it. This is where multi item sorters come in. These systems can sort *multiple* items into one chest, or set of chests. This is usually only done for rare items like ancient debris, or items you simply wouldn't want to have individual chests for, such as the copper variants. You can organise the sorters however you want, to have as many or as few categories of MIS as you like.

Where would I put one?

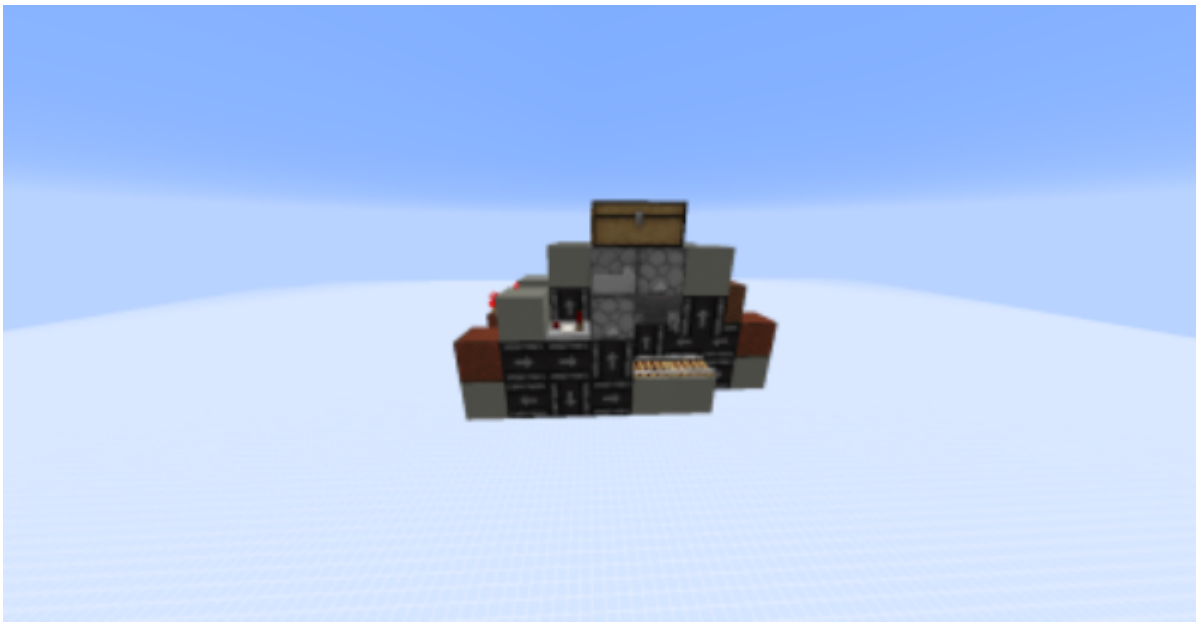
You generally put them at the very end of a water stream, if applicable, as these systems, while powerful, are also slow, only being able to handle one item type at a time, at hopper speed or less. As such, placing them near the end means few items would travel to them and the input buffer would be unlikely to get overloaded.

Some examples:



The "Moony MIS" MIS V5, by MoonPirated and RaPsCaLLioN1138

It can have up to 54 item types per 2wABt slice, configured in the chests near the rails.



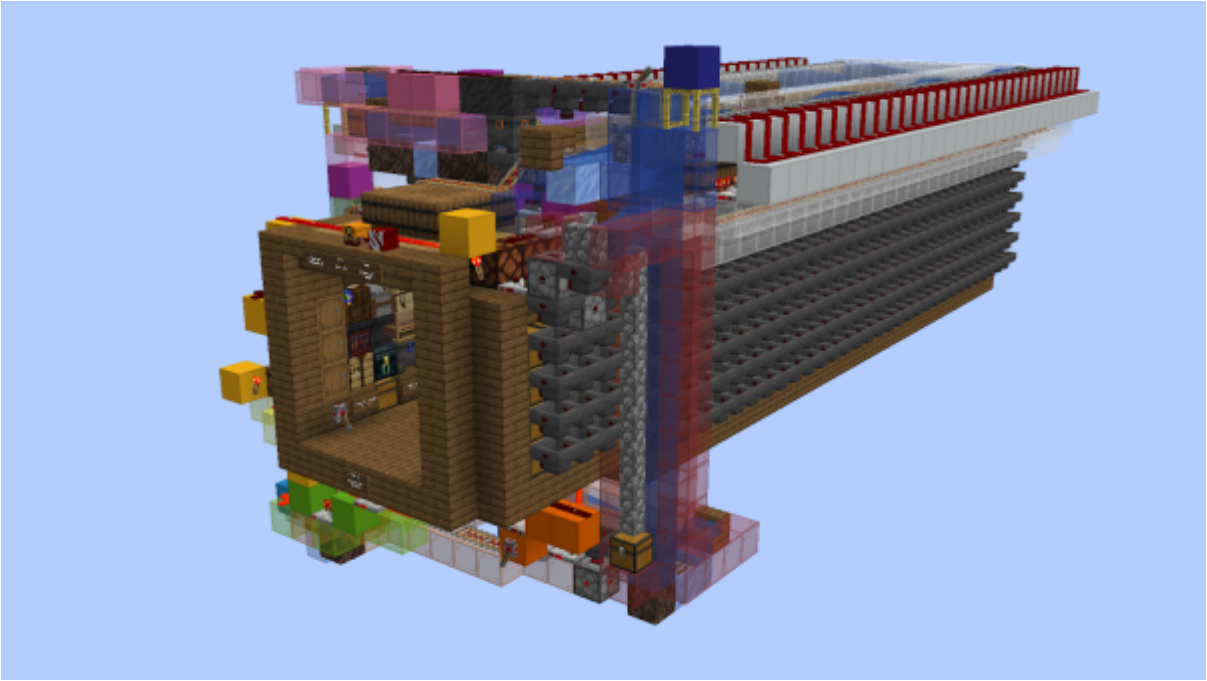
Slot reservation MIS by *Hopper tr-hash order , SamosTheSage , CommandLeo*

This one sorts items into the chest based on if it can push items in, so to configure it you would place items in every slot of the chest, based on what items you want in each chest, and the amount of each
There are other variations I haven't included here.
Some small storage systems rely *almost or entirely* on MIS systems to sort items.



Jayroi's Moony-Item-Sorter (J-MIS) by JayRoi the Hubby, MoonyPirated, SergyD

A system that handles some bulk items, storing them in double chests and boxes simultaneously (a hybrid loader), and everything else in multi item sorters.



The compact categoriser by metamilo, Inspector Talon

A system entirely composed of tiled MIS slices with no dedicated bulk, sacrificing storage capacity for simplicity of building and robustness, and build complexity.

These were the three main components of almost every main storage, but there are some other things to consider.

- Input
- Unloader
- Unstackable sorter
- Chunkloading grid
- Peripherals
- Box loaders

Input: The input module usually is the main interaction between the players and the main storage.



Input System by Soronto

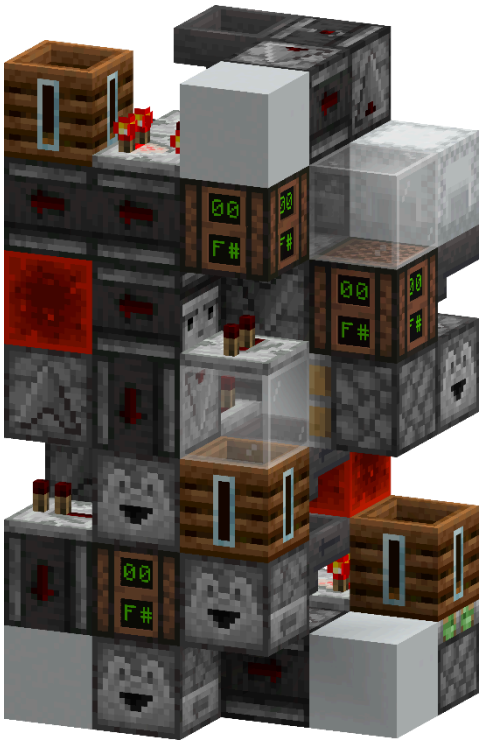
As we see in this example, there is a shulker box, in which we can throw random items we might have in our inventory and want to be sorted and stored. Also, we see two chests.

Each one of these is for different purpose. The first one, is dedicated to bulk input. This is a term that means exclusively boxes full of the same item. The point is to throw for example, seven boxes full of cobblestone, three boxes full of dirt and five boxes full of moss, and they will go directly at the box sorters we talked before, bypassing the time-consuming process of emptying etc.

The other chest, is for mixed boxes, for boxes that aren't filled with one item, but they might contain the spare parts of a building for example, or the loot of an exploration, etc.

At this point I should note that not every storage or input has a dedicated bulk input, because this depends on the rest of the storage, since the dedicated bulk input is going together with the box sorted bulk, if you don’t use box sorters there is no point on using bulk inputs.

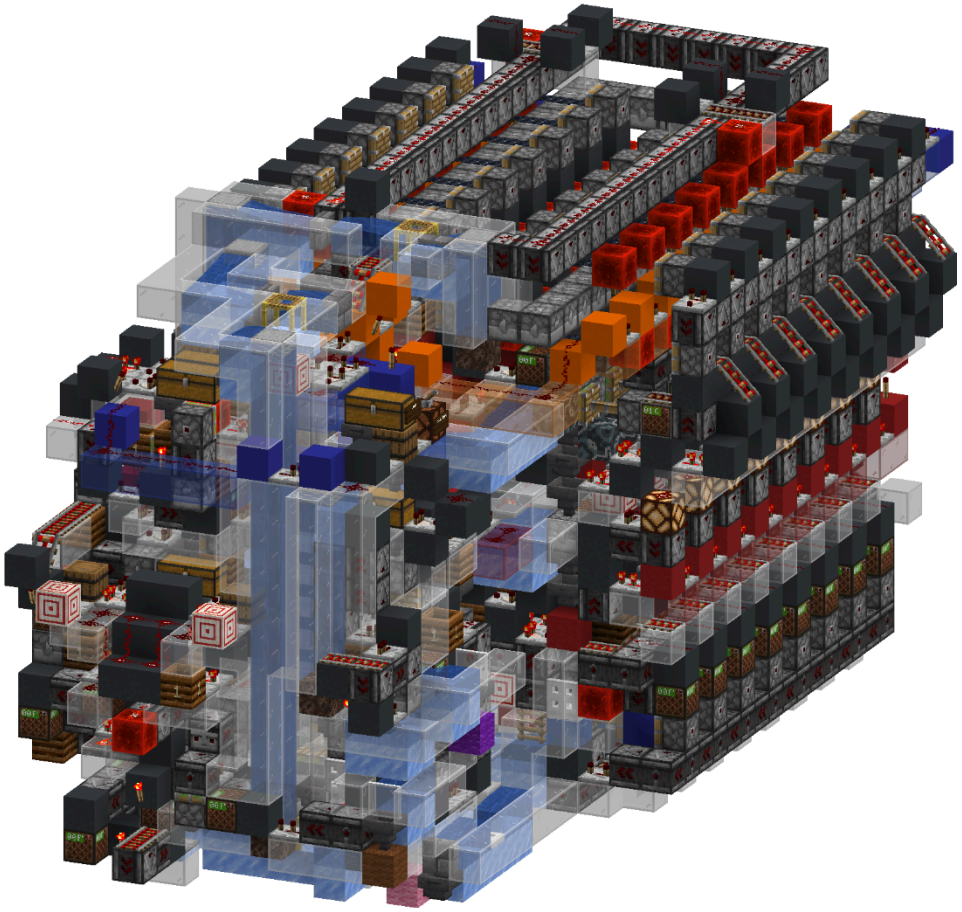
A parenthesis here, since there is always a factor of human mistake, if you want to reduce the possibility of mishandling the bulk input, you can add a box processor that can separate the empty and full boxes from the rest, and this gives you the option to send to bulk only the full boxes, like the below:



Signal Strength Box Fill Level Sorter by 金合欢酱(acaciachan)

Unloader

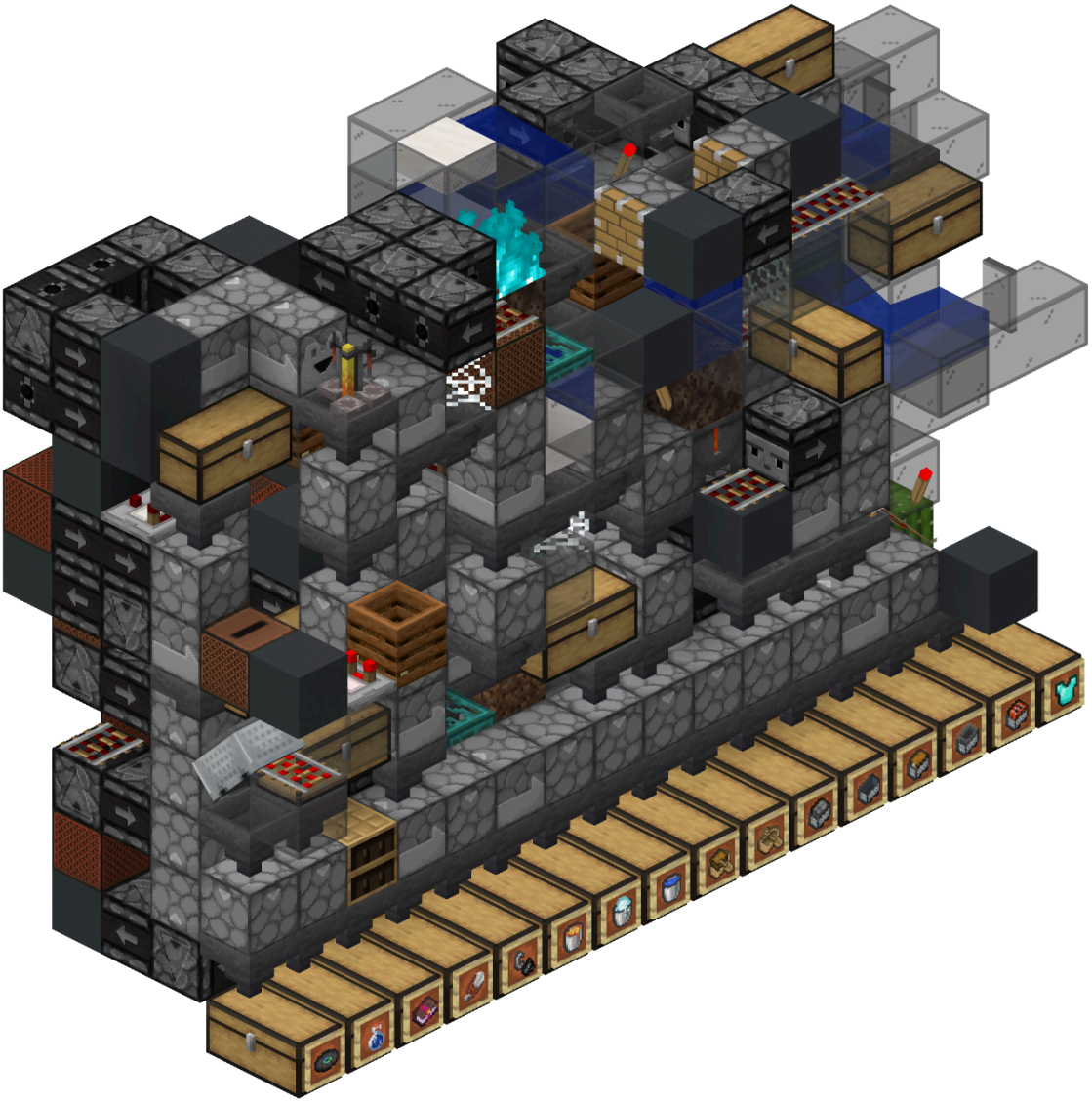
The unloader is one of the components that tends to make the most difference in the efficiency of the storages. It’s a very sophisticated component that allows the mixed boxes to unloaded in parallel, but with a way that won’t clog the system. To explain this with simple words, lets say that we have some boxes that contains cobblestone, stairs, dirt, buckets of water, and a totem of undying. The unloader’s job is to split the buckets of water and the totem in a separate water stream, the shulker boxes in another, and the stackable items in batches of 20 or 60 (depending on the design), all these in parallel to achieve the maximum efficiency. The below unloader can unload items up to 16 different items.



Dueling Hopper Set Based Unloading Array By Kayzm

Unstackable sorters

These devices are kinda special, because they use the properties of some unstackable items like for example the fact that potions and books can go into brewing stands and chiselled bookshelves respectively, to separate each one. As we see in the following sorter, it can separate a lot of unstackables into categories.



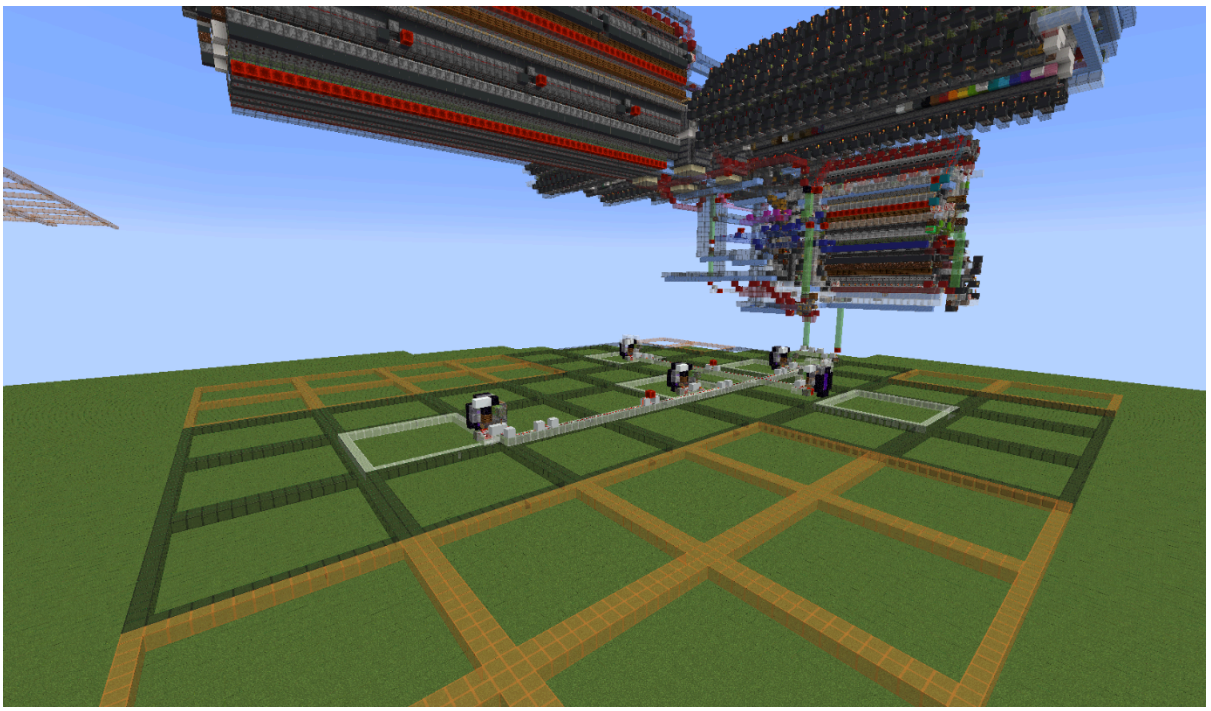
Unstackable Sorter for 1.20 by CommandLeo

Chunkloading grid

The chunkloaders is something we need to consider, because they allow the storage to operate without the presence of a player. This means that we can drop our items and continue to our next destination without the need to stay while it works. Here should be noticed that this wont work if we exit the single player world or the server restarts.

We can also combine our chunkloader trigger signal with a pulse extender and make the chunkloaders operate only when we throw items, and this can be achieved by

taking input from a wooden pressure plate after the unloader. This will reduce the idle lag since it won't need to be loaded all the time.



In the image we can see with white colour the chunks that the chunkloaders have build, with grey the entity processing chunks (3x3) and with orange the 5x5, which doesn't really matter since in most cases we need to have entity processing radius in our storage.

Peripherals

Since a storage system has to do with item handling, some people choose to attach small components like furnace arrays, automatic brewers or crafters. This obviously is a big category of available gadgets and needs to be treated carefully because else it will rise the lag of our storage.

Box loaders

If we want to use a bulk system that have box sorters, we must convert our mixed boxes to boxes that contains only one item, and they are 100% full. To do this, we

need to take the output of our unloader and channel it in an array of filtered box loaders like the following one

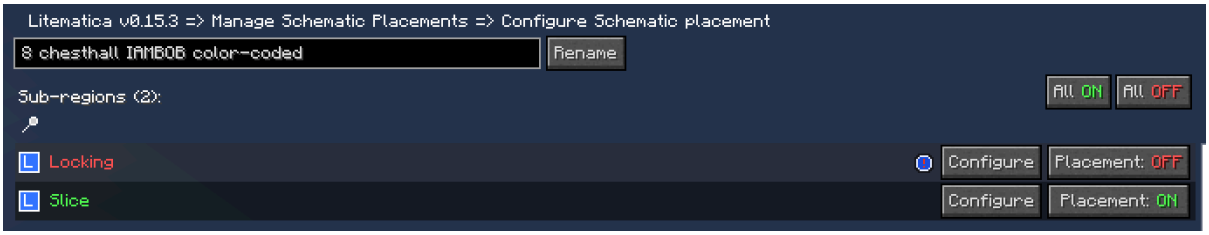


1wt 1x Precision Loader by PyraLeft

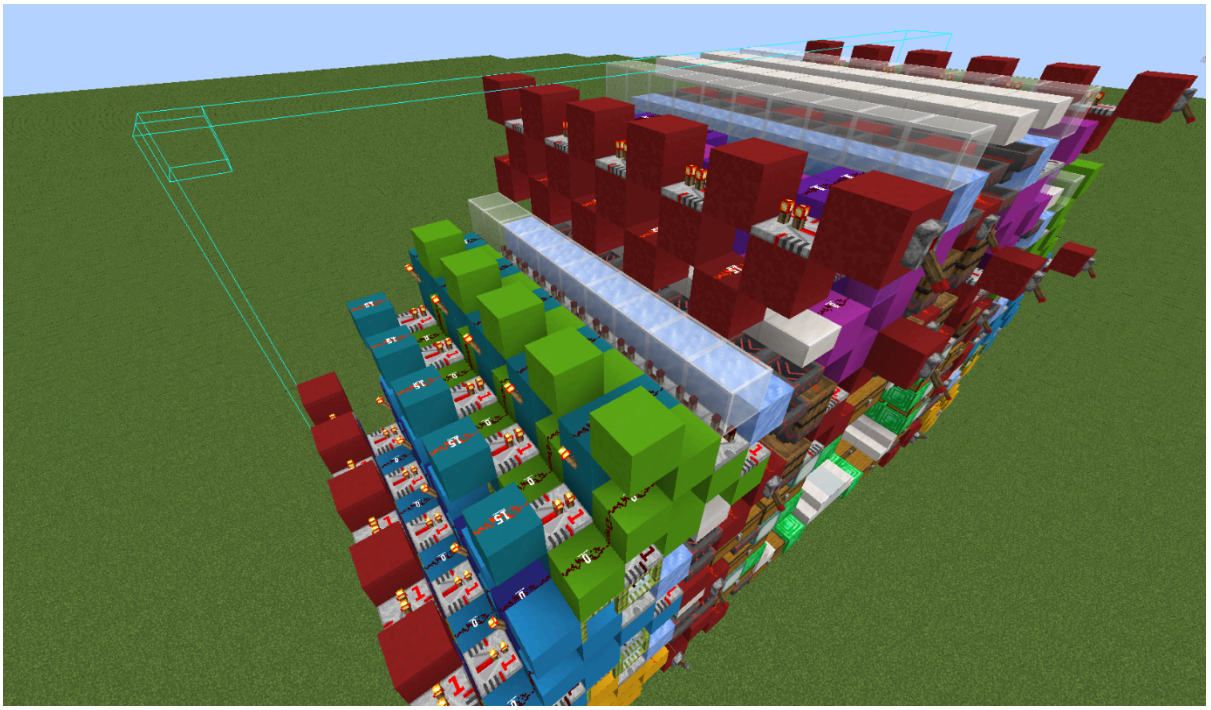
Tilling the halls

In order to make the storages easier to make and more customisable, the biggest part, aka the halls (bulk and chest) and the box loaders are repeatable patterns of the same circuit. From now on, they will be called slices. This comes with some extra terminology, like the 1wt that means 1Wide Tilable, the circuit is 1 block wide and you can tile it as many times as you want without having issues in operation. But sometimes you can't have stuff that are exactly the same next to each other. Then comes the ABt, that means that the design has two patterns, pattern A and pattern B that alternates. This is important to keep in mind when you are tilling. An example for AB tilling is the chest hall design we saw at the beginning.

Something important to keep in mind, when we are tilling, we must take care to not tile the levers or any other indicators. This is noticeable by the fact that they stick out of the design's width, and sometimes (but not always) it might even be in a separate sub-region, like in the following example:



This is how the hall will look after we tile the design 5 times:



The same logic applies also to the loaders, but in the bulk systems with box sorters you should be careful regarding some stuff:

- Some sorters have parts that are ABC tilable, and not AB
- Due to the special nature of sorters, the first 2-3 slices are also special, so don’t tile these, tile the next ones in a pattern like: 1,2,345,345,345 etc

Water streams

Due to the parallelization capabilities of the unloaders, is wise to use water streams to transfer the items from the unloader to the halls. Any other option like for example a line of hoppers, not only removes the parallelization and caps the transfer rate to only 9000 items per hour, but also is extremely laggy and above everything, more expensive.

The water streams must be careful oriented in a way that aren't too long so the items despawn, but also must be efficient because this reduces the item lag. This means that if you have a main storage with 2 chest halls, the correct way is to pass by the top row of both halls, and then goes to the lower level. If you do it differently and the items are going up and down, you are losing time on these transitions which lows the efficiency and increases the lag.

In corners the correct way to turn the items is this:

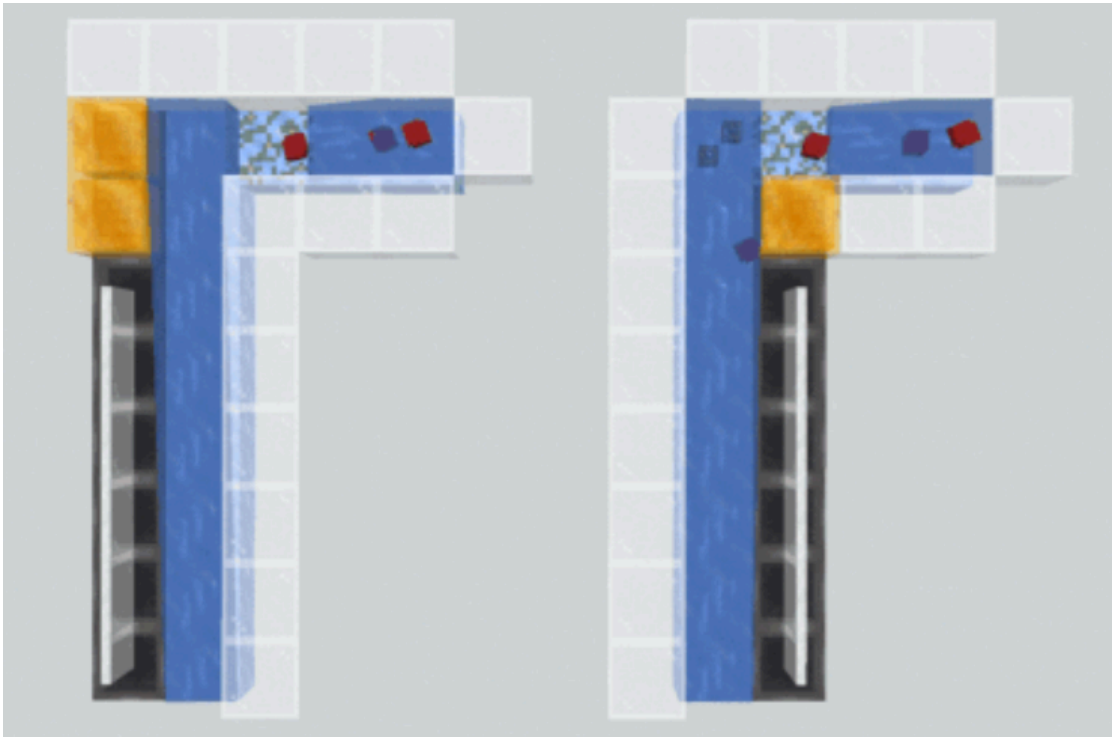


Image by Kartoffel

Keep in mind that the most efficient priority for water streams is:

Unloader -> Bulk loaders -> chest halls ->mis

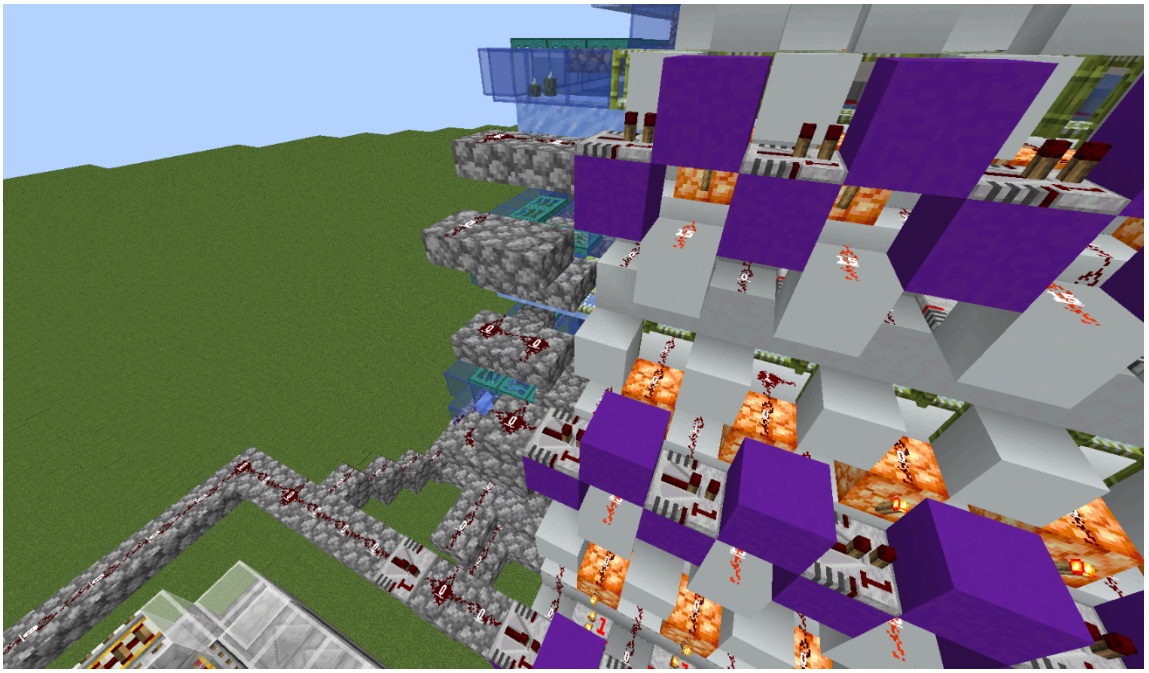
Unloader -> unstackable sorter

Hopper Locking

Its time to do something about the levers we saw before.

Their role is to keep the filters locked when the storage isn't working and thus reducing the so-called idle lag, since every hopper produces lag even if it's not working, and since in the main storages hoppers are very common, we need to do something about this. The levers shows that the block in which they are attached, must be powered or depowered for the filters to work. The specific state, however, depends on the design of each slice, and there is a big variation. That means that some basic redstone knowledge is required to define which "levers" need to be unpowered and which powered.

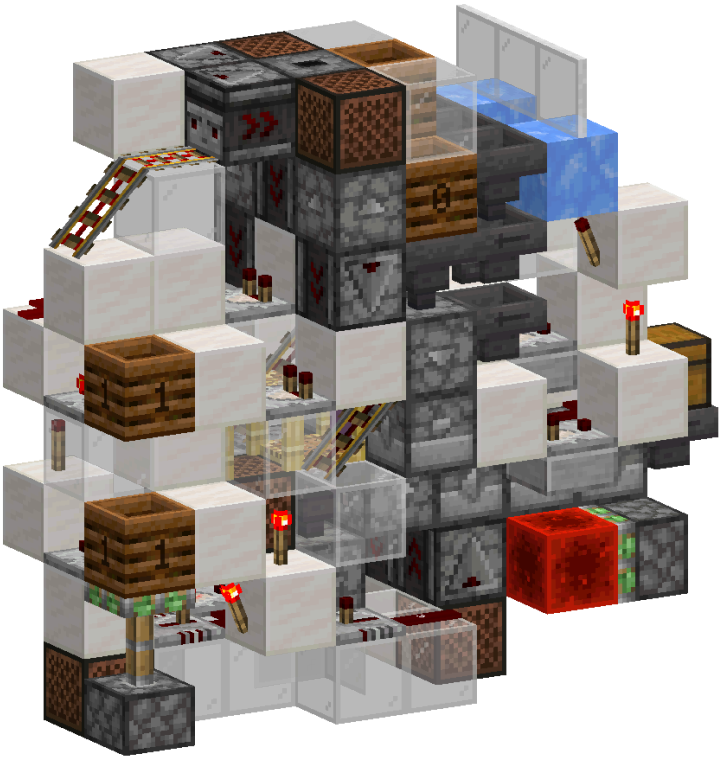
The easiest way to handle the locking is to wire it up in a main locking line, that connects with the input or the output of the unloader, aka the start of the sorting process. That way the halls will automatically unlock when the sorting process is starting. To stop however, you need to measure the time needed since the last item is ejected from the unloader, until this item is sorted and convert this time into a pulse extender.



An ugly example of the main hopper locking line and the slab-snake to reach the correct blocks

What comes after

By experience, there are always items that wont sorted, like for example renamed items, items with custom nbt data, or even rare items that you might decide that you don't need a slot for them. For these cases a mixed loader can be used. Its job is to load all these random items in boxes and delivery them in a dedicated chest, accessible to the user.



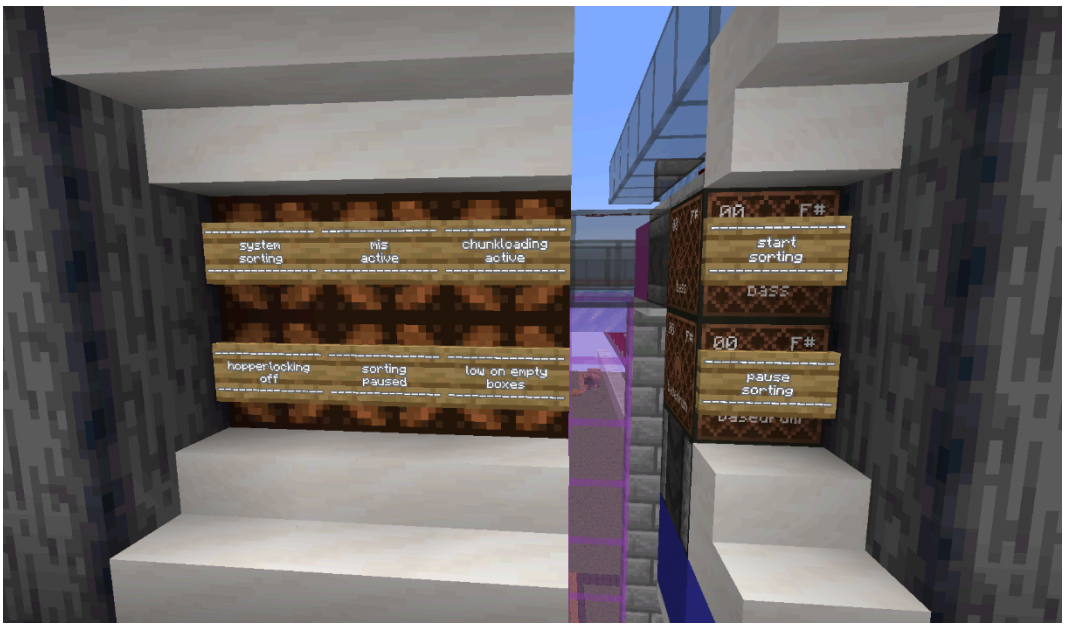
3wt Full Clearance Mixed 6x Loader with all droppers detected by 金合欢酱(acaciachan)

Feedback & control

Although for a bare minimum main storage isn’t required, its always good to have status feedback from the main storage. This means that you can have indicator lamps for when the main storage is working, if the empty box supply needs refill, or if the arrow supply for the unstackable sorter has run out.

Also is wise to add a manual pause for emergencies when you cant wait until the sorting is finished.

Depending on the dedication and effort that is willing to spare someone, there are always stuff to add or tweak.



Yet Another Main Storage (Singleplayer / Small Server Edition) By skyzy and Etikle

Good Luck

Very very thanks to @hiddensquid80 for his contributions