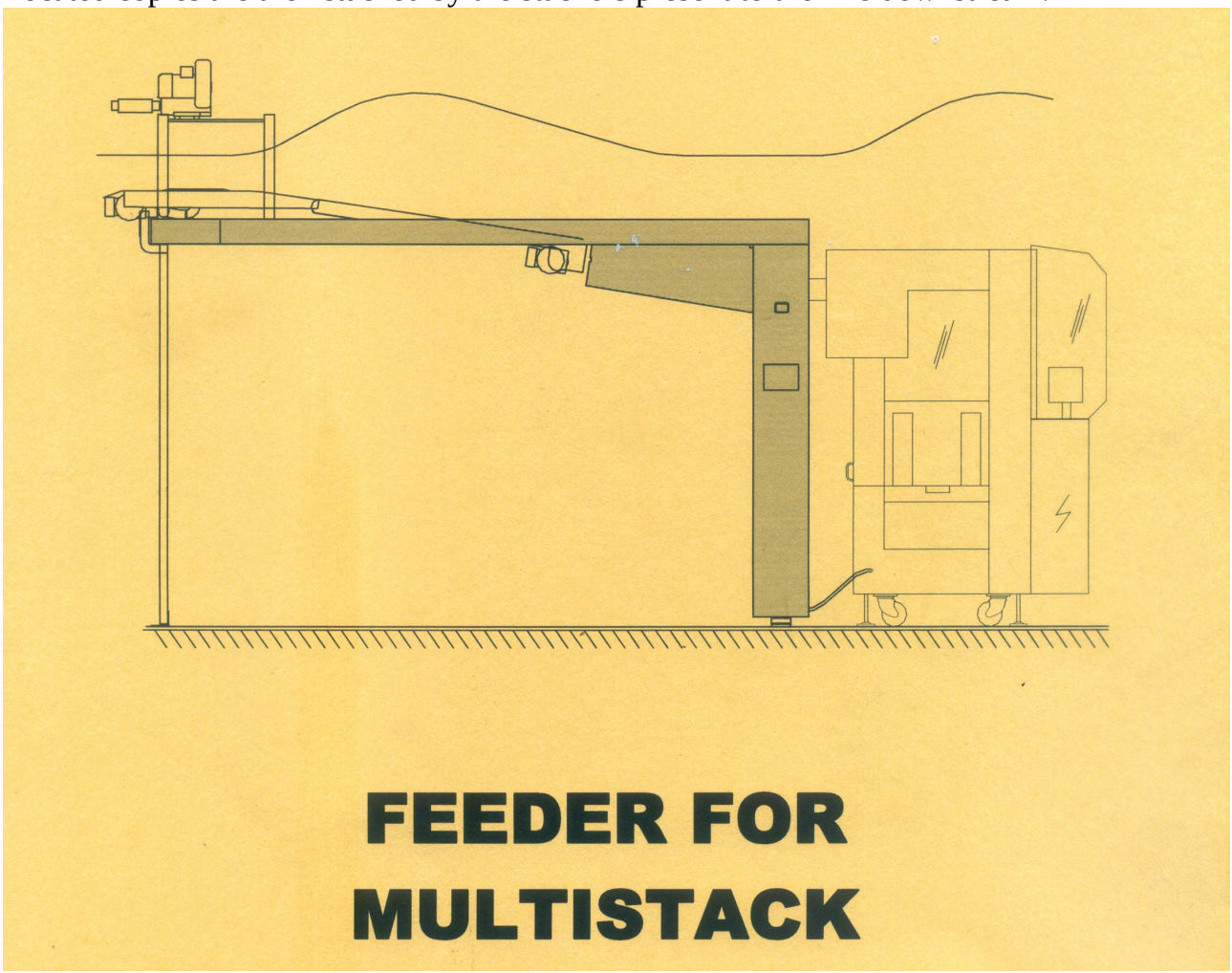
FERAG ALS Modification project in Mailroom – Coimbatore

A detailed report

E. Ranjithkumar

What is ALS:

ALS is copy allocation system which allocates copy to the respective stackers, the allocated copies are then stacked by the stackers present to the line downstream.



We have designed the ALS system ourselves thereby reducing cost to company in present and future from purchasing of spare of around 10 lakhs

BECKHOFF CENTRAL UNIT 2 NOS	= 2 * 2,12940	= 4,25,880
BECKHOFF BRIDGE MODULE 4 NOS	= 4* 54,355	= 2,17,420
BECKHOFF EK1100 MODULE 2 NOS	= 2* 40,216	= 80,432
BECKHOFF HMI ALS 2 NOS	= 2* 200000	= 4,00,000
New type Bridge module firmware updation charges approx	= 70,000	= 70000
Total saved in cost as no need to buy spares stackers as spares and PLC/HMI/MODULES unit can be used in	=	11, 93,3732

PS: for those who don't want to read full tech description there is Tl;dr section at the last page of this document please read that part for the gist of this document.

ALS a brief explanation:

Example: 100 copies per bundle with stack size of 20 is our current production settings

For above settings the ALS system will check if both stackers are in ready condition and allocate 20 copies. If any of the stacker is offline/Jam-up/Error then it allocates all its copy to single stacker, waits for the other stacker to come online and starts syncing both stackers and allocates copies equally to both stackers.

How fast is ALS in Ferag:

case 1:

2 copies/gripper @ 65000 copies per hour.

So for 1 second the system it calculates and allocates totally 18 copies/sec for both stacker time take to release single copy is 0.055 second or 55 millisecond per stacker simultaneously.

case 2:

1 copy/gripper @ 65000 copies per hour.

Time take to release single copy is 0.0275 second or 28 millisecond to stackers.

So within the 0.055/0.0275 sec the ALS system has to read:

1.COPY Detect Signal from denex sensor.
2.Gripper speed reference signal.
3.Relay the copy detect 64 bit buffer data from memory to Release device in both stackers.
4.check Error signals from both stackers and allocate copies accordingly to stackers.
5. Also update the value of the copy present data for each gripper pulse's rising edge.

Within this set time frame, ALS has to do the above calculation accurately, else it may miss the copy counting pulse and copy data may not be updated in the buffer and copy will overflow, since release device does not have the data whether copy is in that gripper or not due to missing that pulse during scan time

All this calculation is done by **Beckhoff PLC** and its data is relayed to the stackers via **Ethercat communication Bridge module,** the bridge module as its name suggest serves as a bridge between both Stackers and the Allocation system, if there is any communication loss between this module then, even if the PLC is working there will be no operation of the stackers via ALS mode.

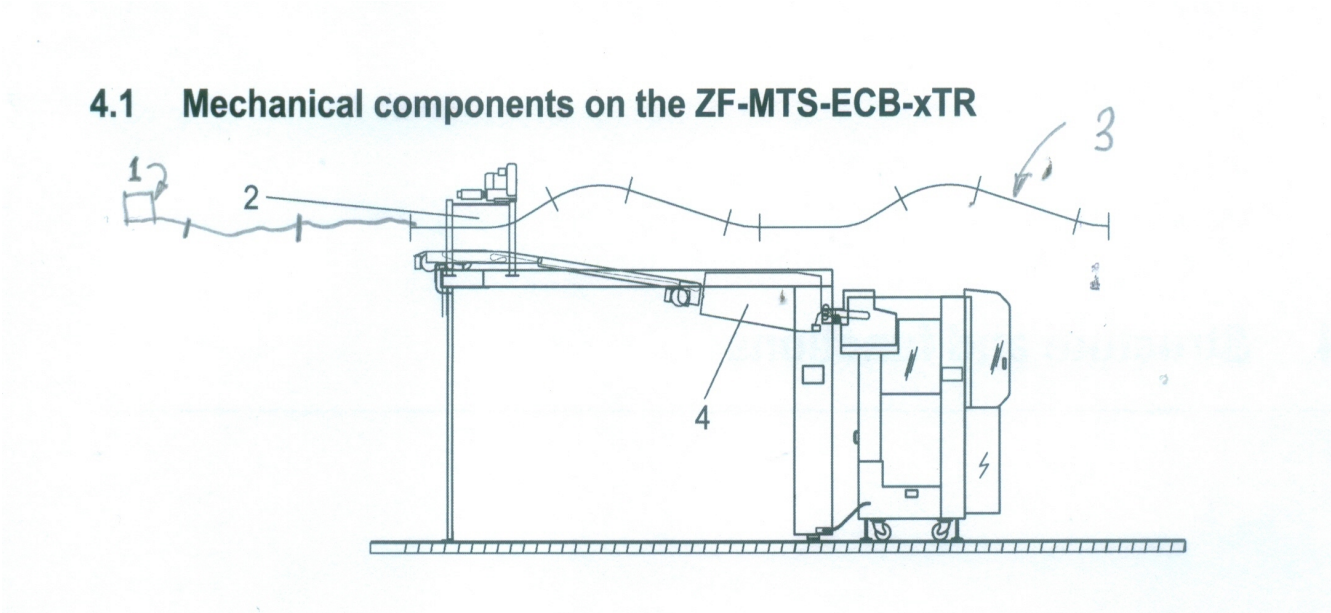
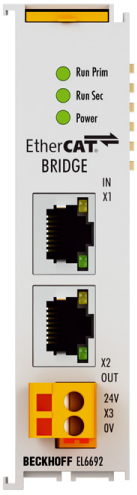


Fig 4.1:
1. Denex copy sensor position.
2. copy release device position1 for S**tacker 1(**23 grippers away from Denex copy sensor), for S**tacker2** Denex sensor is 63 grippers away from release device 2.
3.Copy transport Griper chain (UTR).
4.Stacker copy Infeed area.

<u>EtherCAT BRIDGE: EL6690</u>



NOTE: as of 2021 this module is redundant and Beckhoff has stopped this line of modules, only Ferag is supplying these modules till their stocks last, the Beckhoff has even removed this module from their website, even stock image is not available for them.

The Problem:

ON 2/09/2020, the Ferag ALS system has stopped working with the error “Bridge not ready “so we were unable to run machine in ALS mode, so we quickly changed the production in **Standalone mode,** and ran the edition, after checking we have come to the conclusion now that the bridge module is defective since its run led is not lit. when we replaced it with the spare module available with us still same problem persisted. we will give the steps undertaken to approach this problem briefly below.

The Temporary solution:(running stacker in **Standalone mode**):

With **Standalone** mode only one Stacker is online all the times, and the other Stacker is in standby mode, if any error occurs in the Online stacker after a brief overflow, the Stacker 2 will come online from standby position till the Stacker 1 resets. Then stacker1 will be in standby and the other Stacker2 will be in online mode, so effectively without the calculation and allocation by the ALS, stacker cannot function to its full potential, the release device will be always on and all copies will be diverted to whichever stacker is online.

Even though the PLC module is working, without it passing on the gripper copy data via “Ethercat bridge module 6690” to downstream machine, the stacker cannot function at its full potential.

Problem when running in Standalone mode:

Since only one Stacker is in online mode and other is in standby mode we simply cannot go high speed printing beyond 40k copies/hour, there is frequent jam-ups, copy bundle creation was not good, copy separation in the fork-loop area is also not good, so high speed printing is not an option now, this aspect seriously affect us because, a single delay in production can have a large cascading effect.

So we made some minor modification in the available machine wiring and fixed a timer for releasing device of both stackers now both stackers are in online mode, but still copy allocation to each stacker is time bound not counter constant, so copy steam is not exactly 20/50 copies per stack hence fork-loop jam-up as in old standalone method, though there was slight improvement as stacker burden is reduced now. But still bundle formation was not good and there were many chances that copies per stack may vary by 2 copies extra in one stack and 2 copies less in the next stack due to fork-loop separation issues.

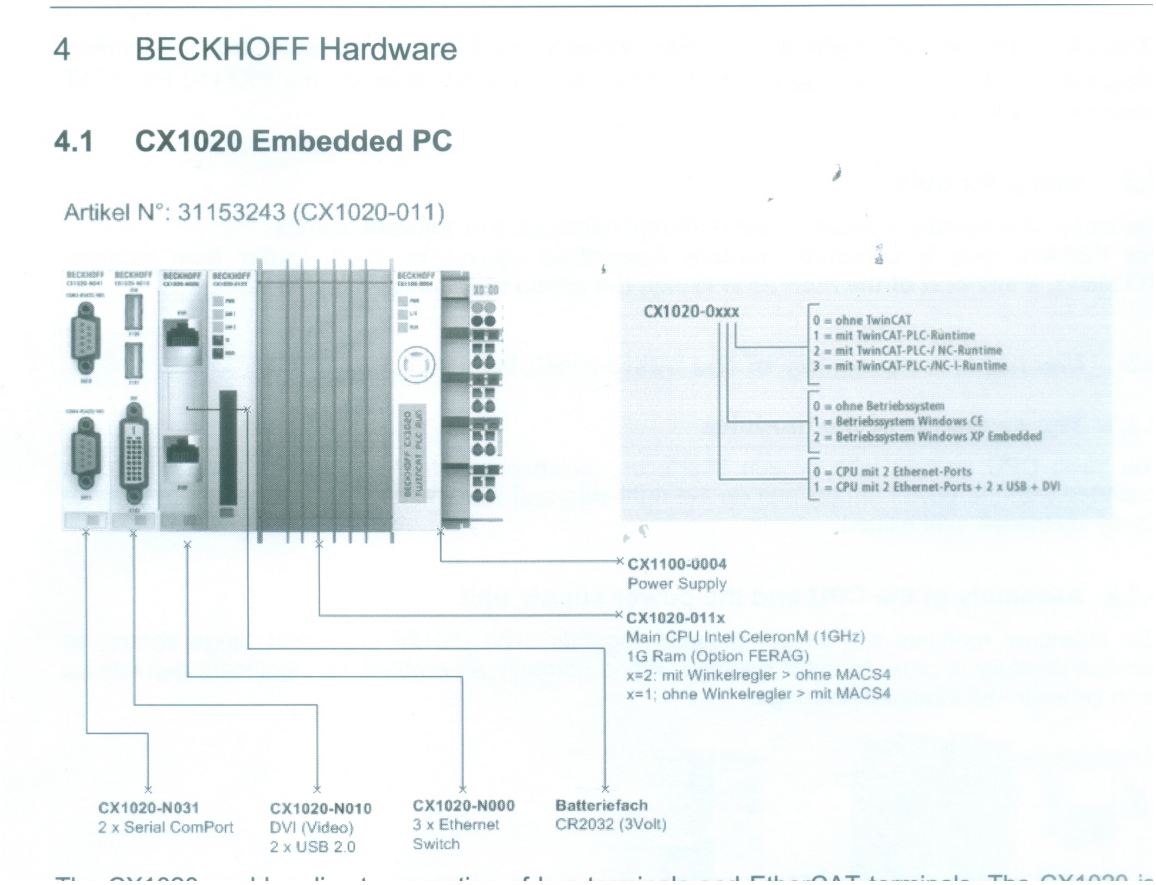
We tried all the available option with us, so we tried contacting Ferag India for sorting out the issue, Mr. Subramani of Ferag India was briefed about the situation and he agreed to remote connect to our Ferag System and try sorting out our issue, after connection he too suspected that our both spare and the working “COM” module has failed simultaneously and he can only proceed after us buying a new module, to reconfirm the suggestion from Ferag India, we took

our module to local Beckhoff they also iterated the same conclusion as the Ferag India. But the catch is Beckhoff has stopped this line of product and only Ferag has limited stock and they advised us to buy new model of this “com module” also they have to come and make firmware modification which will cost us 50/60k extra approx. we knew this process would take more than two months so we thought why not try designing the process system as whole

There are 2 Ethercat bridge module(one for each stacker) per ALS, each costing 50k Rs each, also any single point failure is collapsing full ALS system unusable, so we thought that if there is any possibility of designing this whole ALS system ourselves from the scratch, please do note that unlike MHI, Ferag doesn't share its PLC code since its their IP so we have to design it from our working knowledge and intuition alone.

THG ALS system designed and implemented successfully in house

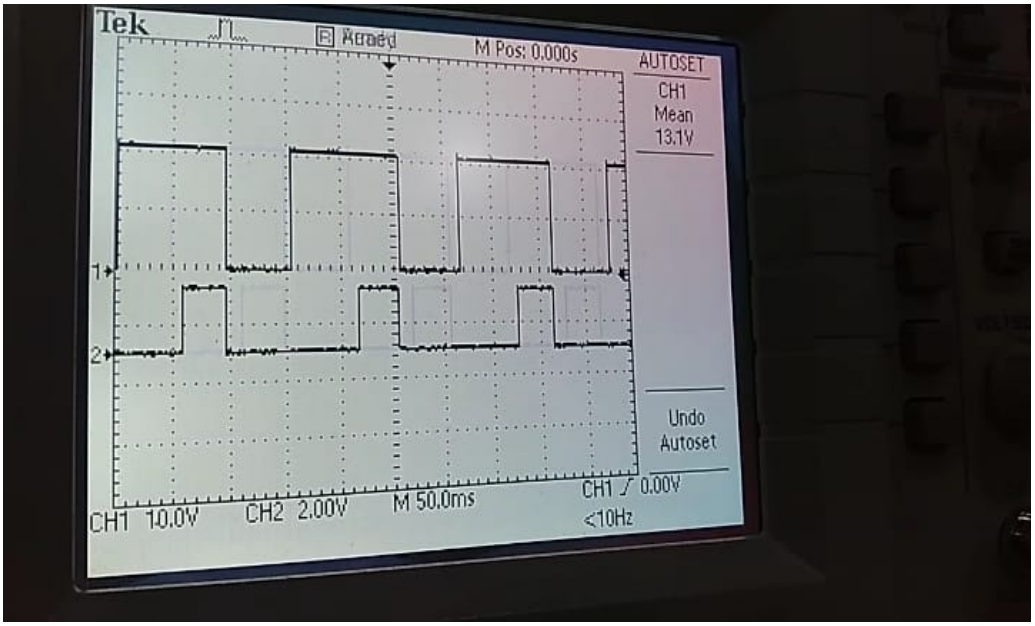
As mentioned earlier in this write up, we only have schematic diagram as program logic was not given to us since Ferag considers it as their IP property. Also since Beckhoff system used in Ferag is getting old and becoming legacy so we have decided that we have to implement this system as whole including PLC LOGIC. A thorough study of the hardware wiring was done and with that data we tried to come-up with a solid logic for programming further.



We will explain further how we in-house created a rugged program controller, replacing heavy duty industrial Beckhoff IPC/PLC, We have 2 Beckhoff PLC,4 Bridge module,2 Hmi touch panel totally in both ALS. We don't have to buy the spare for the ALS in future also which is a very big plus. Cost saved breakup is in last page for reference.

HOW THE LOGIC WORKS:

As soon as the gripper starts it creates a 64 bit serial buffer, the program scans for the input from the Denex copy counter and Gripper speed pulse signal and be in ready mode



In the oscilloscope we can see the both denex copy detect signal ch1 and gripper sync signal ch2 observed @ 40k speed, see the 50ms sync pulse, if we miss this pulse during scan cycle, the copy release data will be missed and copies will go to overflow and counting/syncing problem may/will occur. so prudent use of good programming practice must be used, else this pulse which gets even smaller with high speed will be missed.

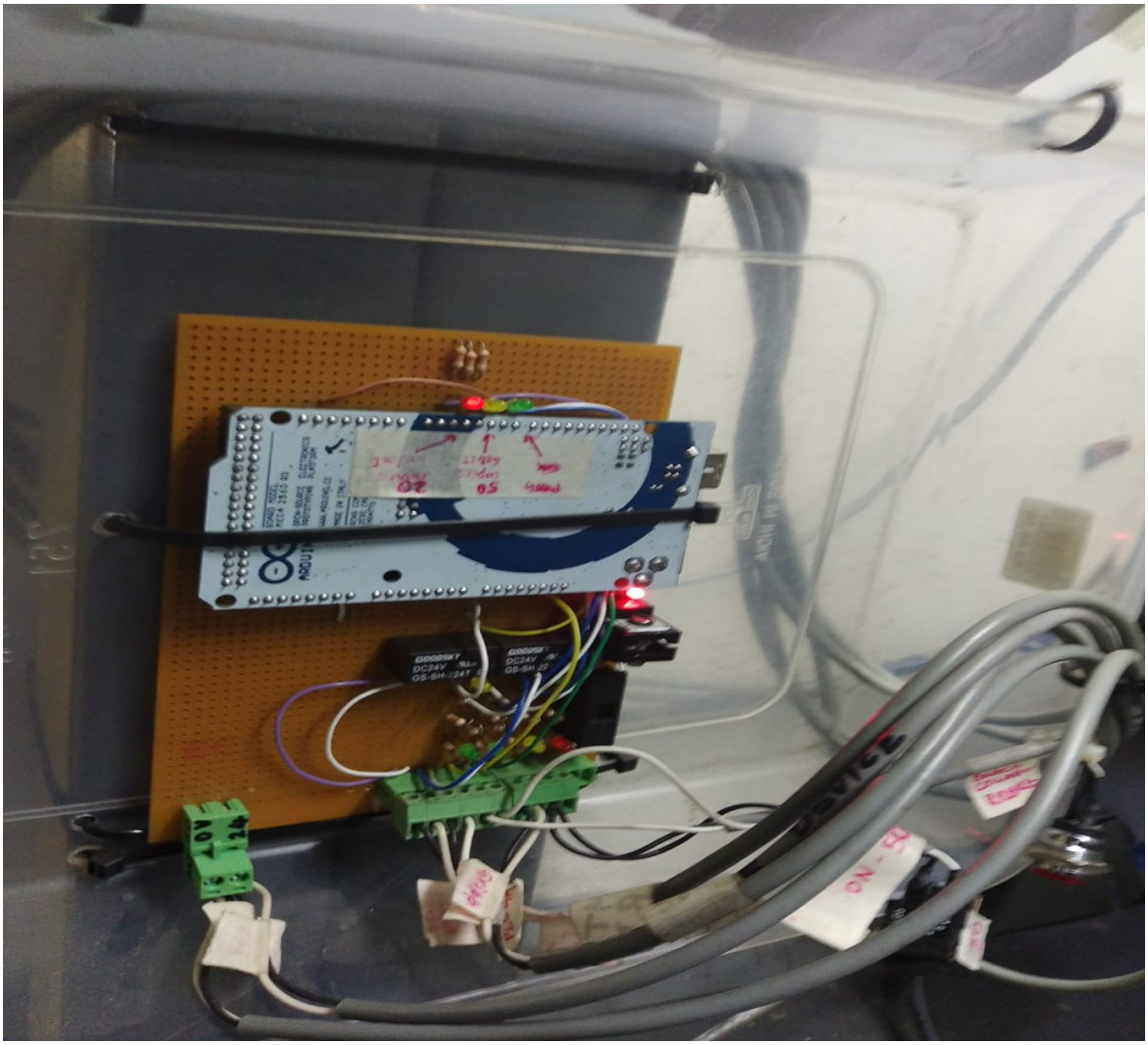
If any copy detect signal is present in the gripper its value is updated in serial buffer and its data is passed on the Copy release devices of both stackers located on the 23 and 63 grippers away from the denex sensor. On each rising edge pulse of gripper + High of copy detect signal we update the copy present in Griper “memory buffer” and pass on to the release device and start two counters one for each stacker.

Now the controller will start an 8 bit register, which holds how many copies both stackers need, number of copies presently released, copies pending per stacker and syncs both stackers & allocates 20/50 copies according to the production.

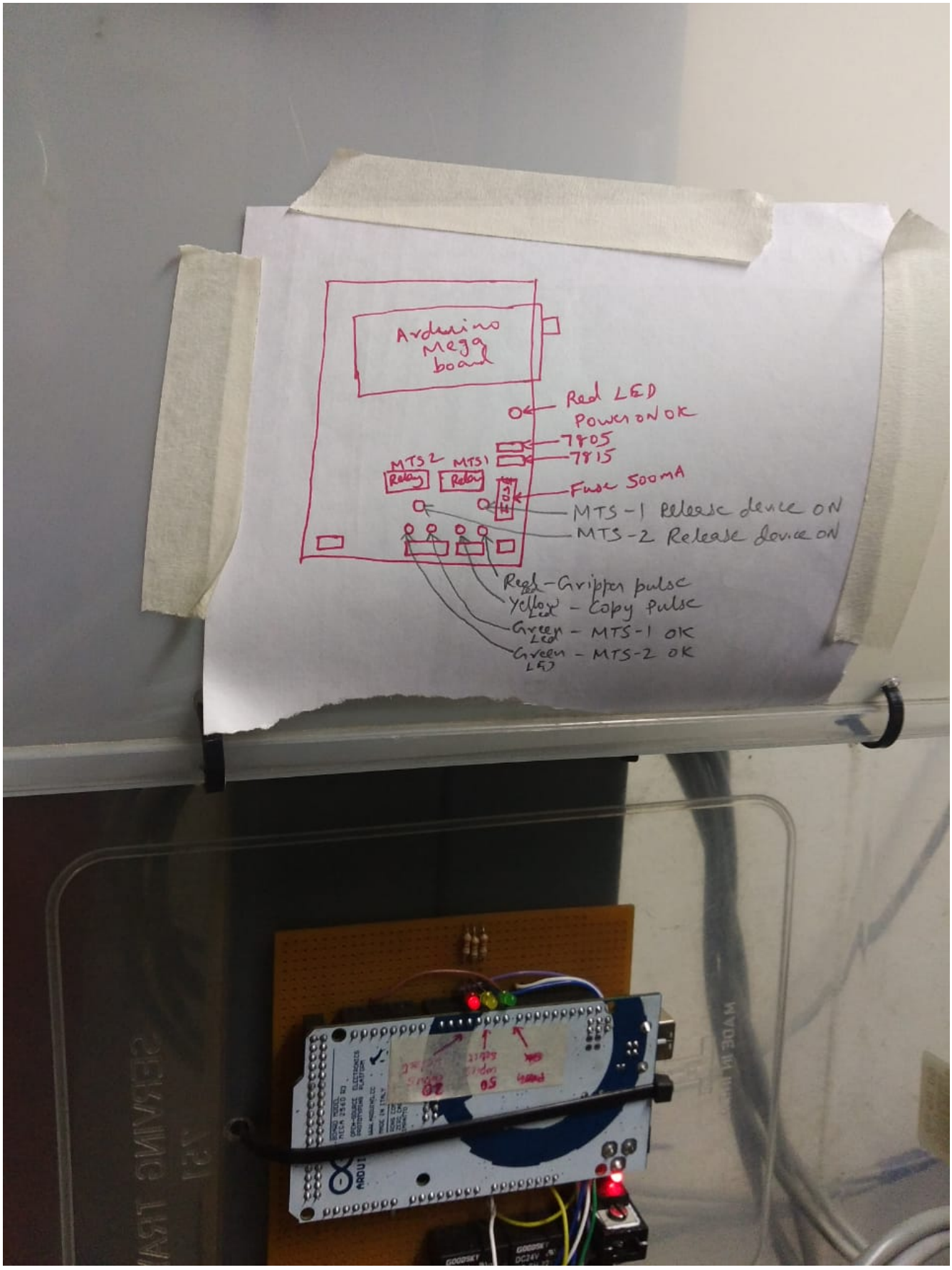
This is what happens in ideal settings but we are not in ideal world, so if any problem/error arises in any of the stacker, the copies which were being allocated to the error stacker will be diverted to the stacker which is still online, since the program stores the number of copies being released it never looses the count of the copies to be released to each stacker and when error stackers comes up online after error reset, it knows how much copies both stackers need to re sync itself.

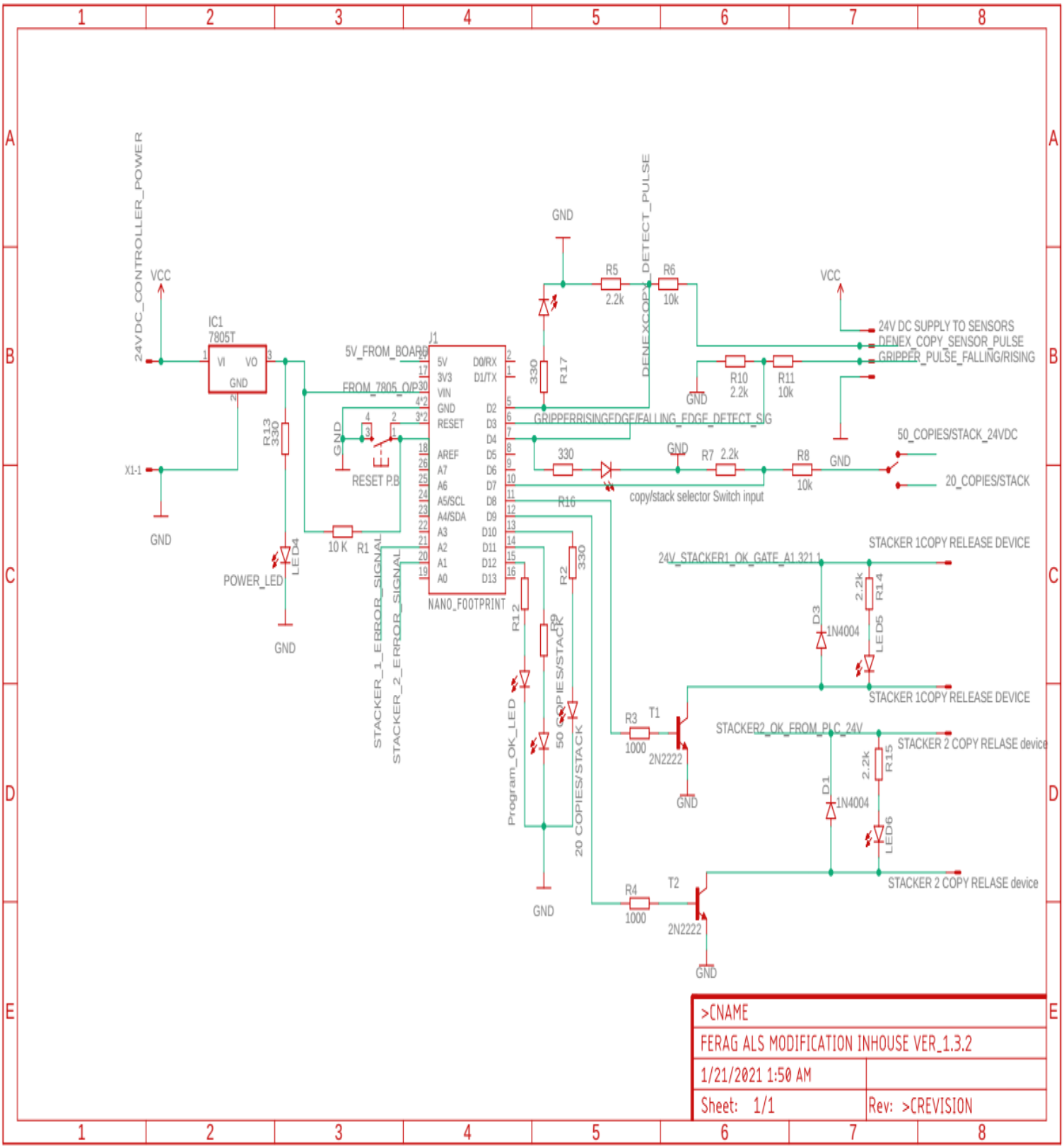
After creating the programming logic in Embed C, we tested it in slow speed -feeding the copies to the gripper manually in pickup station 300 copies at a time,”every time’ we need to test, then observe copy detect and copy allocation process, we had to go back to the drawing board redesign/rewire/reprogram the required parameters many time and it caused us great difficult and frustration at time, at-last we persevered and came out successfully designing the ALS system, which is running for three months successfully without any problem, so we stared designing this for another ALS system present is DS folder line also, saving tens of lakhs in the process, now we may no need to import the modules that has become costly, old & legacy, requiring the attention of OEM, if we need to replace it with their newer module then firmware update is mandatory they may have to come visit for the process and charge us extra.

We have achieved with few hundred of rupees what a PLC which costs lakhs of rupees does, we were feeling very happy and ecstatic that we could pull this off, that too on a control process which requires very fast processing. We have gone further and above and tested our system to respond to a speed pulse as short as 15 millisecond(0.015 seconds).



The above set up is temporary test setup running successfully for 3 months, we have started/going to complete the wiring in cabinet for a permanent setup shortly


Arduino Mega board
Red LED Power ON OK
7805
7815
MTS 2
MTS1
Relay
Relay
Fuse 500MA
MTS-1 Release device ON
MTS-2 Release device ON
Red - Gripper pulse
Yellow LED - Copy pulse
Green LED - MTS-1 OK
Green LED - MTS-2 OK



schematic diagram: of the in-house ALS system implemented using ATmega2560 Micro controller

THE ALS SYSTEM WHICH HAS BEEN REPLACED BY US:

COST BREAK UP:

To immediately run the machine in ALS mode we need Bridge module since we have designed it ourselves there may be no need to purchase this module again since the implemented system works on par with the original Ferag system, the central PLC unit which is used in ALS can be used in both stacker as a spare, same with the Hmi panel and EK1100 EtherCAT module

BECKHOFF CENTRAL UNIT 2 NOS	=	2 * 2,12940	= 4,25,880
BECKHOFF BRIDGE MODULE 4 NOS EL6690	=	4* 54,355	= 2,17,420
BECKHOFF EK1100 MODULE 2 NOS	=	2* 40,216	= 80,432
BECKHOFF HMI ALS 2 NOS	=	2* 70,000	= 4,00000
New type Bridge module firmware updation charges approx	=	70,000	= 70,000
Total saved in cost as no need to buy spares and PLC/HMI/MODULES unit can be used in stackers as spares	=		11,93,3732

Note: The price break up is a year old, it may have increased from the above mentioned value.

TL;dr:

Single point **failure in ALS copy allocation system in Mailroom Ferag** has rendered the whole setup unusable and causing production delays, also all the spare used in the machines has come to end of the cycle, they have also increased cost of those spare and announced last buy option, so we created our our **in house** designed and developed **system** and saved significantly cost to the company (around Rs 9,33732 approx)in present as well as future.

THANK YOU,

FROM
E.Ranjith kumar
TEAM COIMBATORE.