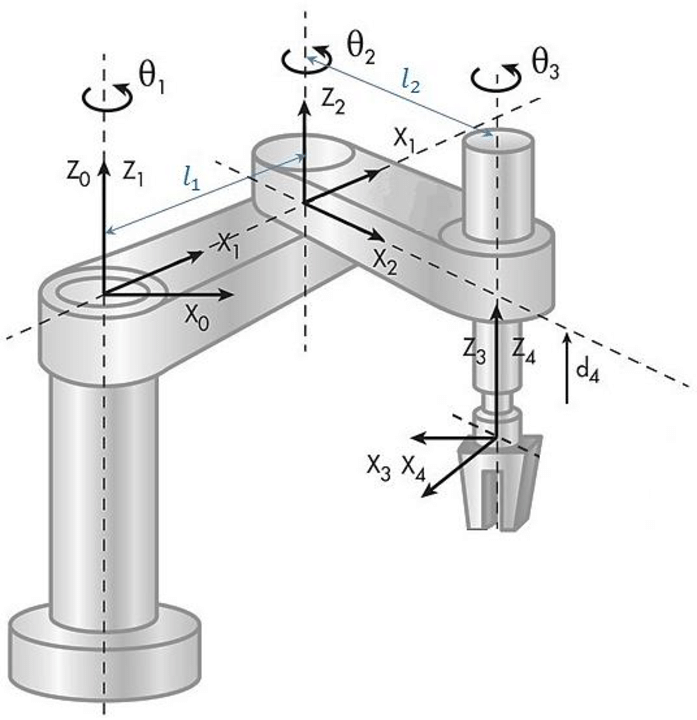
****

**REPORT-1 JULY 2021**

**What is SCARA?**

The SCARA is a type of industrial robot. The acronym stands for Selective Compliance Assembly Robot Arm. Selective Compliance Articulated Robot Arm, meaning it is compliant in the X-Y axis, and rigid in the Z-axis. The SCARA configuration is unique and designed to handle a variety of material handling operations. Articulation is the type of motion and degrees of freedom required by the application. For a robot, the number of joints, joint location and the axis each joint control determines its degree of freedom. SCARAs are four-axis robots, with motion in X-Y-Z and a rotational motion about the Z- axis. Most pick and place and small assembly applications that involve moving a part from point A to point B are perfect for SCARA robots because this typically involves relocating the part in X-Y-Z with some rotation in the Z-axis. In addition, for dispensing applications, a SCARA can pick parts and present them to a dispenser, or the dispensing tool can be attached to the robot’s shaft. The work envelope or the area of space that a robot can physically reach is a critical consideration. Typically, SCARA robots have a cylindrical shaped work envelope with variations in the diameter and depth of the cylinder. In most applications, a SCARA’s work envelope is contained to the front and side. The back area may not be usable if cables and pneumatic hoses exit from the rear - some SCARAs offer optional bottom exits – making it possible to work behind the robot. Speed is an important factor when choosing a robot, and SCARAs are typically one of the fastest on the market. With four axes, they have fewer moving joints. SCARA printers differ in mechanical nature from other FDM printers as their motion system involves the use of multiple controlled arms. These arms enable the printer to move the printhead around the X and Y planes. The printer’s arms typically use two motors on the same side that are connected through joined arms. This arm motion system incorporates linkage to cover the full range of the print area. The firmware of a SCARA printer calculates how the motors should move and in what combination to have the printhead move to print the desired part. This arm motion contraption moves up and down typically using a threaded rod along the Z-axis.

****

**Advantages of SCARA robots**

* The SCARA robot is most commonly used for pick-and-place or assembly operations where high speed and high accuracy is required.
* Generally, a SCARA robot can operate at higher speed and with optional cleanroom specification. In terms of repeatability, currently available SCARA robots can achieve tolerances lower than 10 microns, compared to 20 microns for a six-axis robot.

**Limitations of SCARA robots**

* SCARA robots, due to their configuration, are typically only capable of carrying a relatively light payload, typically up to 2 kg nominal (10 kg maximum). The envelope of a SCARA robot is typically circular, which doesn’t suit all applications, and the robot has limited dexterity and flexibility compared to the full 3D capability of other types of robot. For example, following a 3D contour is something that will more likely fall within the capabilities of a six-axis robot.

**Advantages of SCARA in 3D Printing**

* They have large build areas for their footprint. Cartesian 3D printers have to move the head within itself in order to reach all the points on the print bed, where a robotic arm can reach from its perch to a point on a print surface outside its footprint.
* The bed is much larger because it is hanging on the machine instead of sitting inside of it.
* The arm is very light if you can manage to remove the motors from the arm. This allows for neat fast movements without too much vibration. This makes the machine quieter and relatively faster.
* The open architecture allows you to get to the finished product easier. It also makes it easier to demonstrate 3D printing (educational purposes) since it all happens in full view from a 180-degree angle.

**Disadvantages of SCARA in 3D Printing**

* The math is hard and calibration can be a challenge to get right when you develop a new system. This alone will dissuade many companies during the development phase.
* You have to use high gear ratios to get the movement resolution right, although this can be done with standard NEMA17 steppers and very large pulleys.
* The openness can be a problem when you print materials that shrink a lot. In that case, you can encase the whole machine, or invest in a hood for the device.
* [Low payload](https://robodk.com/blog/scara-robot/) (the printhead has to be lightweight)

**Commercially available SCARA 3D Printers**

**1. RepRap Morgan**: - This was the first SCARA 3d printer to be built. It is an open-source fused deposition modelling 3D printer. The Morgan is part of the RepRap project and has an unusual SCARA arm design It has a build volume of 380 x 220 x 220 mm and a hot end that can heat up to 245 degree. The RepRap Morgan was the first SCARA 3D printer to be built. Multiple versions of it have been made.

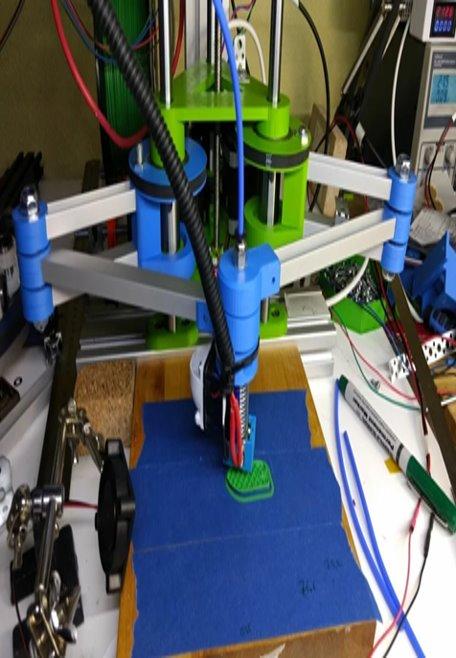
This printer has won multiple awards like Humanity Plus Uplift Personal Manufacturing Prize and third place in the Gauteng Accelerator Program.

**Features:**

* Print Head: Single extrusion
* Print Head temperature: Up to 245°C /473°F
* Maximum Build Volume: 380 mm x 220mm x 220 mm
* Heated bed option available on request.
* Allows printing of many more filament types including
* PET-G
* ABS
* Nylon
* Flexible filaments
* Layer Thickness:
* 70 microns | 0.07 mm (High Quality)
* 200 microns | 0.2 mm (medium print)
* 300 microns | 0.3 mm (fast draft print possible)
* Internal Storage: 16GB – if configured mounts in the operator’s PC as an external drive
* Weight (without spool):  12 kg (subject to change)
* Dimensions
* Height: 650 mm
* Width: 400 mm
* Depth: 500 mm (Base 350mm

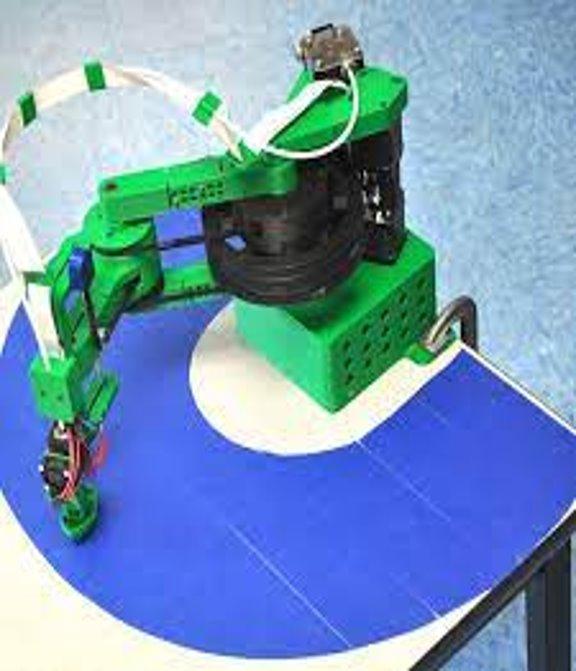


2.**Five bar parallel SCARA:-** this printer is made mostly of 3d printed parts. It uses 12 mm rods ,bearings,belt and 3060 extrusions. It can print very quickly up to 300mm/s with an acceleration rate of 12000 mm/s^2. The controller board being used for this 3D printer is Duet3D,which has a powerful 32-bit 120MHz ARM Cortex-M4 microcontroller. Due to its unusual parallel arm design, it runs on custom built firmware as the duet firmware was not compatible or configured for this printer.



**3. RepRap Helios**: - RepRap HELIOS is a 3D printer that uses a SCARA arm while at the same time has no moving steppers. It is tiny, light, quick, accurate, and has a massive print area. This printer has a build volume of 200 x 200 x 100 mm and also has auto bed levelling and an LCD panel. It runs on smoothie ware firmware. If we add some wheels to this printer then we will have infinite build volume, we can print objects bigger than the printer itself but there is a limit for the height of the object to be printed​.

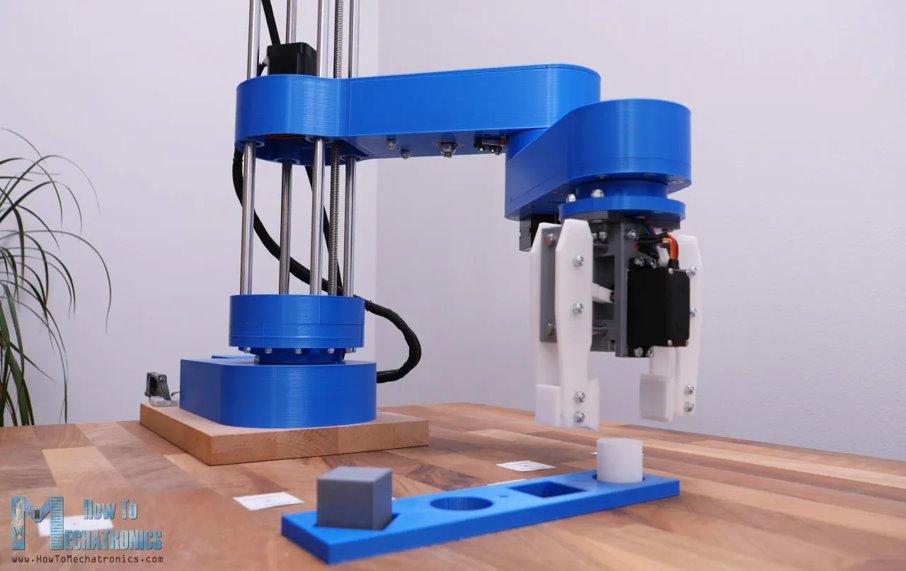
**Features**:

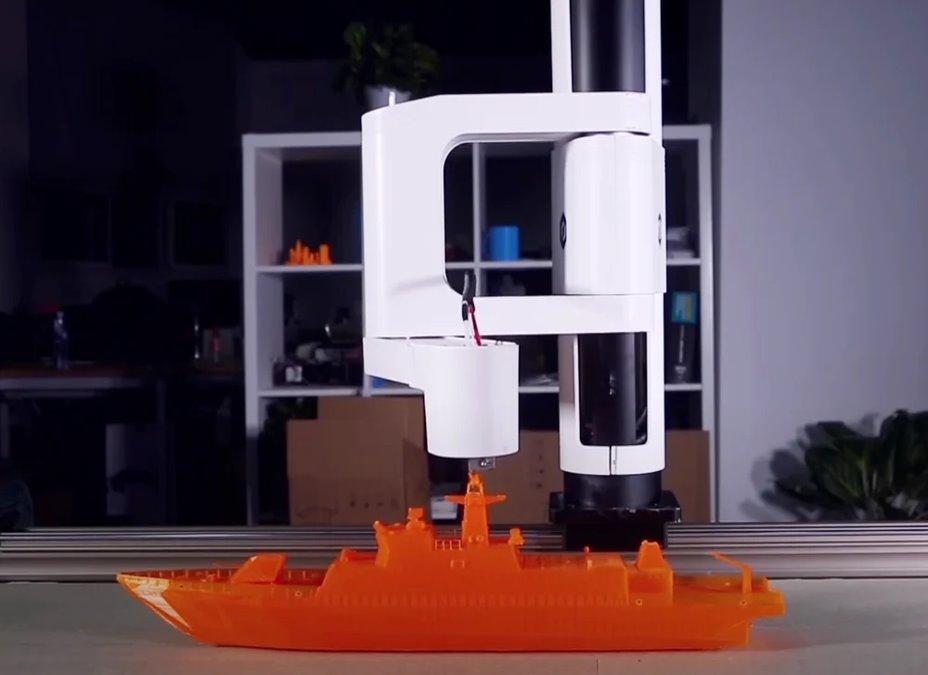
* The design is parametric so all of the details can be modified.
* Build Area: 5 times bigger than a 200x200mm printer.
* Build Height: 100mm
* Speed: Prints well at 30mm/s but have printed at 90mm/s with acceptable results.
* Accuracy: Calibration cubes have been shown to be square in all dimensions and dimensionally accurate +/-0.4mm
* Precision: It is a SCARA so hard to say and microstepping helps but can't be fully trusted. Let's just say at full steps it is good enough and in practice it is awesome.
* Auto Bed Levelling: Yes
* End stops: No, uses a magic accelerometer to fully characterize the machine.
* Controller: Smoothie ware compatible 

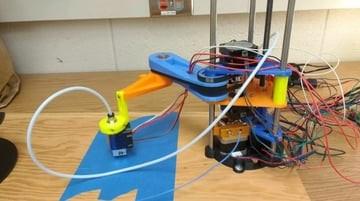
**4. SCARA Arm**: - It has a two-link arm, similar to a human arm. The end of the second joint can move through the XY-plane by rotating at the base and elbow of the mechanism and an added Z-motion by moving the base of the first arm on four vertical linear rods with a lead screw. This printer has no attached build plate and a minimalistic frame. This has a very high maximum printing speed of 100mm/s and a printing volume of 200 x 200 x 150 mm. It uses a proprietary firmware that isn’t open source.​ The SCARA Arm 3D Printer is a purchasable SCARA printer that has no attached build plate and a minimalistic frame. It has a printing volume of 200 x 200 x 150 mm with a listed maximum printing speed of 100 mm/s. The firmware is not Open Source, it's Not Windows10 compatible.

**Features:**

* All-metal structure, durable;
* The dual-arm bearing structure, high strength, seamless, good accuracy, low resistance, spike all the simple structure without bearing arm;
* Gear - belt transmission, high speed ratio, spike gear motor; Reason: The gear motor is a gear shift, there is a gap of 1 MM gear and shift relatively small.
* Small footprint, large print size;
* On the desktop, such as flat print directly, no heat beds, energy saving, afraid burns.



**5. Dobot M1:** - This is an industrial SCARA 3D printer that does things like 3D printing, laser engraving, soldering, drawing, writing, filing papers, picking up and sorting items and as an assembly bot too. This printer comes with attachments such as printer head, laser module, grabber and more. This allows us to change tools on the machine to perform various tasks. This printer has a reach of 400mm, payload of 1.5kg and 0.02mm repeatability. It runs on Linux software too. DOBOT M1 is a cost-effective intelligent robotic arm for the light industry. With high precision, wide working range, complete functions and secondary development, it provides users more ways to use. M1 can realize multiple functions of assembly line work such as soldering, visual recognition and PCB plug-in, helping to construct the intelligent industrial system.​

**6. MPSCARA**:- The MPSCARA is another 3D printer build available on Thingiverse. MPSCARA stands for “Mostly Printed SCARA”, meaning that most of the components to this printer can be 3D printed. 

**7. 5 ARM SCARA 3D PRINTER :**A SCARA (Selective Compliance Assembly Robot Arm) is a specific kind of robot that is defined by its rigidity in the Z and axis and speedy movement in the X and Y axes. All of the joints pivot on the XY plane, and only the base moves up and down in the Z axis. This particular robot design works in the same way, but with linkages between the XY motors and the end effector where plastic is extruded. By pivoting the first two arms of in the linkages, the exact position of the hot end can be set in the XY plane. Movement in the Z axis is similar to a conventional 3D printer, and is provided by a lead screw.

The primary advantage of this design is that it can print very quickly—up to 300mm/s and with an acceleration rate of 12,000 mm/s^2. The quality isn’t amazing at those speeds, but is quite good at around 120mm/s, which is still faster than most Cartesian FFF (Fused Filament Fabrication) 3D printers. The controller board being used for this 3D printer is a Duet3D, which has a powerful 32-bit 120MHz ARM Cortex-M4 microcontroller. The Duet3D has official support for SCARA 3D printers, but this unusual parallel arm design required custom firmware. For most people, the benefits of a build like this aren’t worth the challenges. But it is still fantastic to see people experimenting with unconventional 3D printer designs.

**MODEL1**

The machine is made mostly of printed parts and items you can easily get hold of, and a hacksaw and hand drill. No machined parts. The precision is not the best but it still works pretty good .It's still a prototype/proof of concept so it does not look very good and many parts are stuff I had laying around.

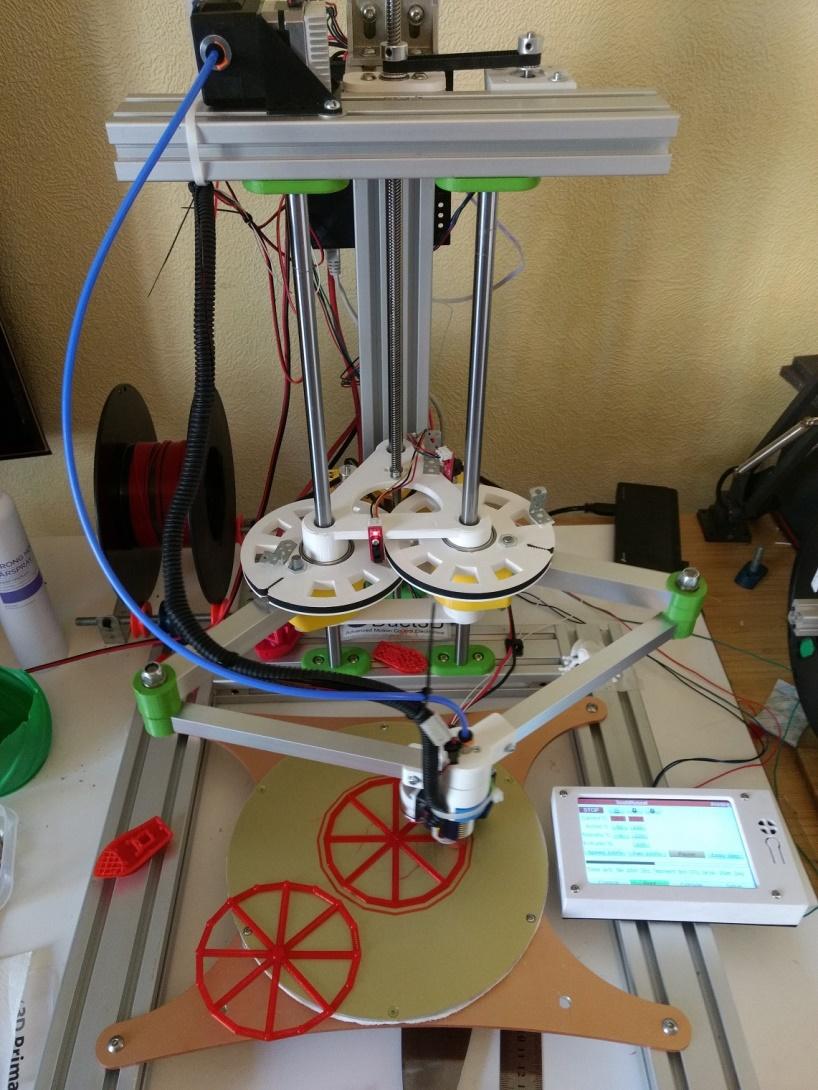
The body rides on 3 12mm rods with LML12UU-like bearings, and has one leadscrew. The inner joints have two 6806 bearings per arm. The elbows use two 8mm thrust bearings each, preloaded with an m8 rod, a crude solution that works very well. That joint is very stiff. The hotend joint has two 6704 (20x27x4) very thin and light bearings inside. Lots of bearings.

The motors driving the arms are 48mm 0.9degree NEMA 17 with a 20t pulley leading to a 200t printed pulley. This gives a gearing of 1:10 which is very much on the low side. 1:30 would be far better. But upping the microstepping to 64 or even 256 saves the situation. A problem with 256 microstepping is that the Maestro starts to get stepping rate problem over 400mm/s.

Using a normal gearbox is not possible due to their backlash. A machine like this needs a gearbox with almost no play, like a harmonic drive. A belt gearbox works pretty good but the gearing is limited due to size.

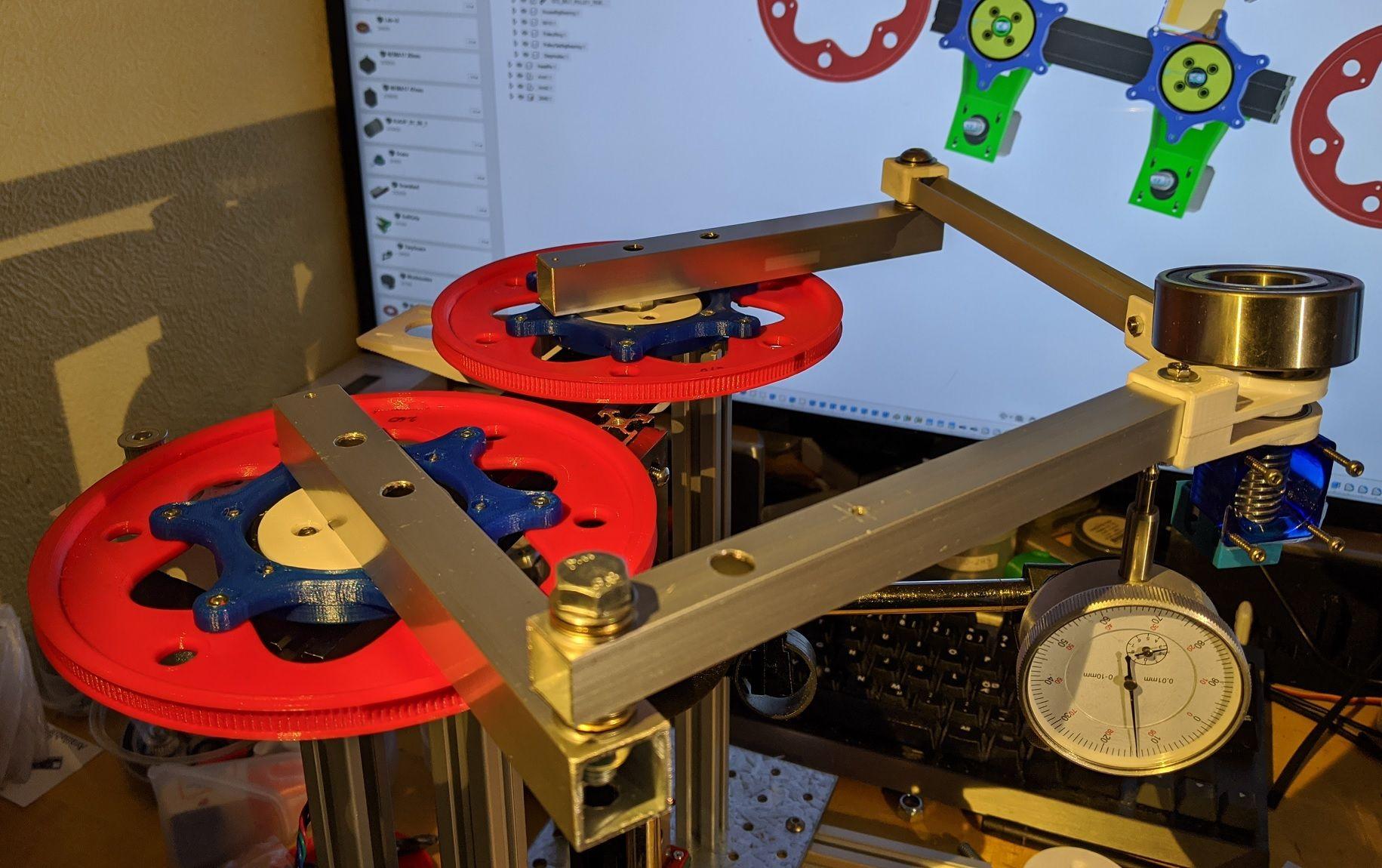
With the 180 and 216 mm long arms the printing area is approximately 300x200mm. The actual print area is actually half-circleish shaped and quite a lot bigger.

The printer can move at blazing speeds and accelerations without any drama and shaking. The kinematics is similar to a polar delta but just 2 axis instead of 3. Currently it has a fair amount of ghosting at high speed/acceleration, but prints very good at a more reasonable 60mm/s and 2000 acceleration. I think the ghosting is due to a far too wobbly tower, or possibly flex in the belts. I have got some metal clamps for the 12mm rods and some more 3060 extrusion in the mail, it might help.



<https://www.youtube.com/watch?v=MW8HApFoy38>

**MODEL 2**

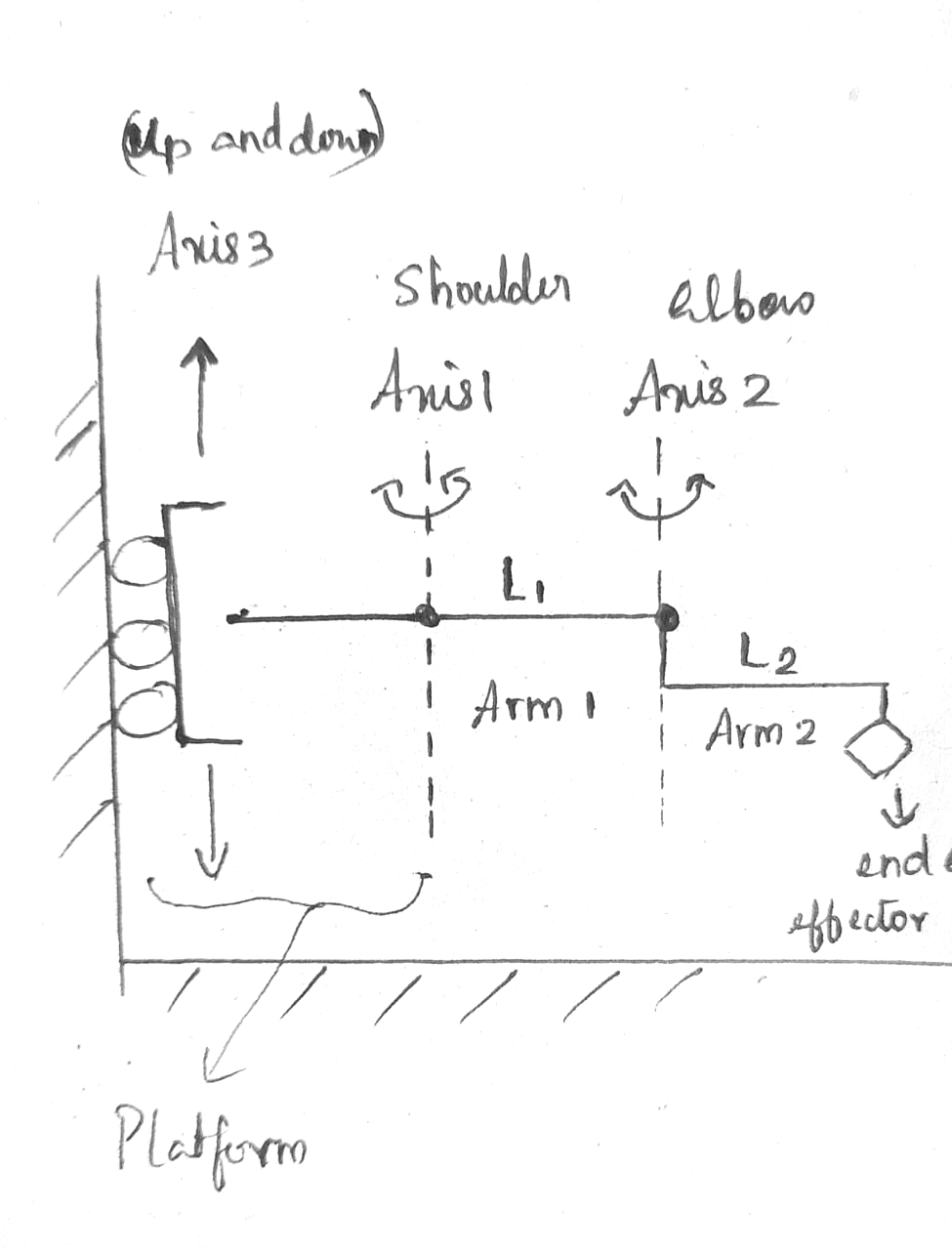


<https://www.youtube.com/watch?v=yubS3_OUhQs>

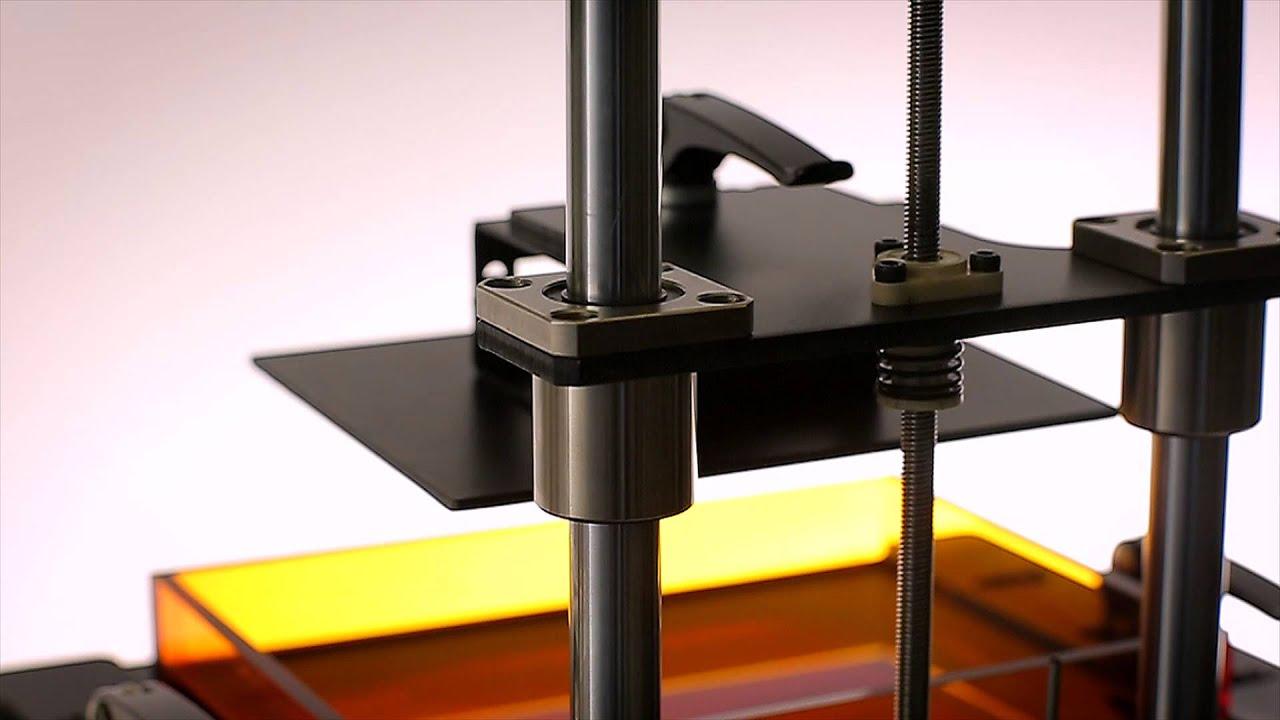
**Mechanism of Single arm SCARA robot**

The degrees of freedom of the arm of the SCARA 3d printer is 3.

* The entire arm can move up and down along the z axis
* Shoulder joint(S) and elbow joint (E) enable motion of the end effector in the XY plane.



* At the base of the printer, rods which act as tracks for the motion of a platform up and down along the Z axis are attached.
* This motion is made possible using a lead screw and stepper mot
* The rotation of the motor shaft is transferred to the lead screw either directly or using spur gears and belt in between.
* As the lead screw rotates, the platform moves along through it.



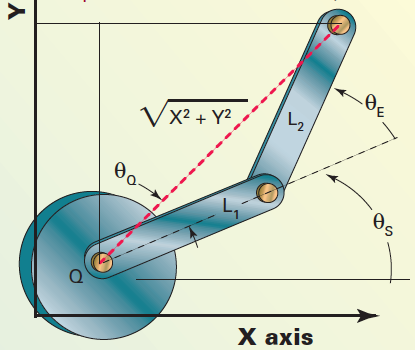
* One end of arm 1 is attached to this platform in such a way that it rotates about a perpendicular axis passing through the platform. Consider this joint as joint 1 and the axis as axis 1.
* This rotation is enabled using a stepper motor mounted on the platform and spur gears and belts (no. and size of the spur gears can be determined based on reduction in rotational speed required.)
* The belt drives can be replaced by direct drives, which eliminates the problem of belt ageing, stretching or cracking and consequently increases precision and eliminates the need for maintenance and regular belt change.



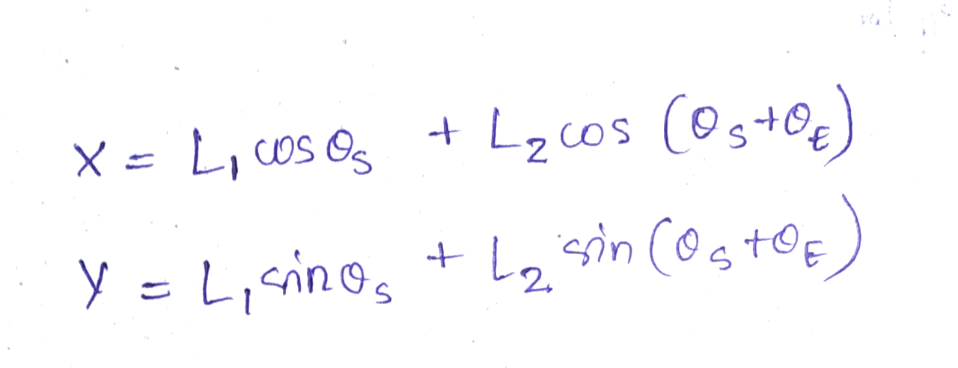
* To the other end of arm 1, one end of arm 2 is attached such that a revolute joint is formed between them (Axis parallel to axis 1). Consider this joint as joint 2 and axis as axis 2.
* To enable the rotational motion at this joint a stepper motor can be placed on arm 1 and a similar mechanism of spur gears and belts can be used as earlier.
* Another way is to place this stepper motor too on the platform itself and transfer motion from there using spur gears and belt through joint 1.
* The extruder is fixed in the other end of arm 2.
* Therefore, this robotic arm has a cylindrical workspace with

Radius = Length of arm 1 + Length of arm 2 (Maximum horizontal distance the arm can reach from the base)

**Control of position and orientation of the arm: -**

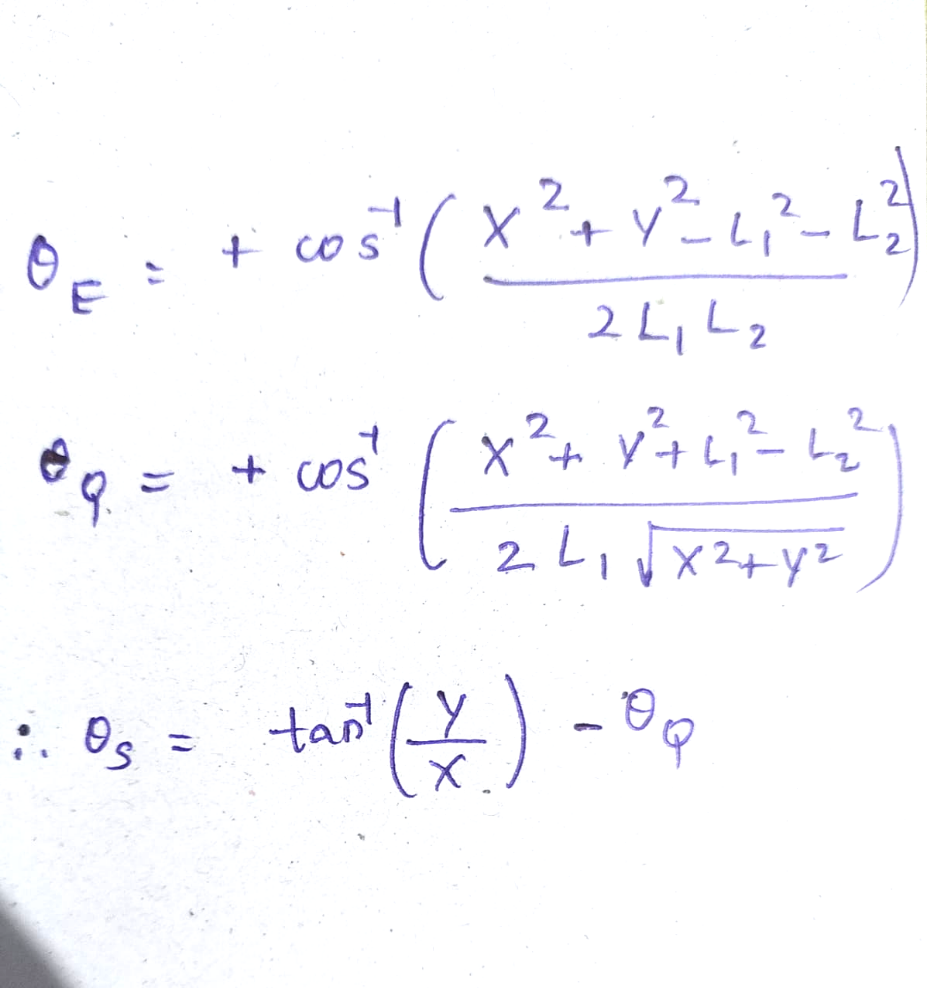


**1)Forward kinematics**

The position of the end effector is found out from the joint angles.

**2)Inverse kinematics**

The joint angles are found out using the position of the end effector.

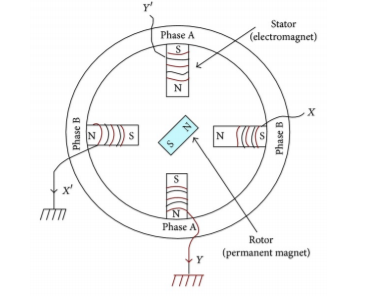


SCARA stands for Selective Compliance Articulated Robot Arm. They are robotic arms that have a versatile range of motion in the X-Y plane

**Components needed :**

1. **Stepper motor**

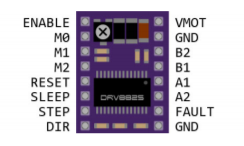
A stepper motor generally consists of a rotor that is a gear shaped permanent magnet which is surrounded by the windings of a stator. The windings are alternately powered to incrementally rotate the rotor on which the shaft is attached. This results in the ability to precisely control the angular position of the shaft



A two-phase PM stepper motor circuit schematic diagram

1. **Stepper motor driver**

Due to the fact that the powered phase must be alternated to rotate the shaft, control electronics that allow for rapid changes in direction and amplitude of the current in the windings are necessary. Such electronics are known as drivers and are available in many different forms.

DRV8825

1. **Bearings**

Bearings are machine elements used to keep components such as a fixed axis in place while also relieving its load. An often desirable trait of a bearing is its low friction, allowing the connected component to move with ease. Bearings are classified by their allowed range of motion or in what direction the load is being applied

1. **Guiding rods**

The main function of the guiding rods was to distribute the load and serve as a rail for the whole arm to move along Z axis

1. **Micro Switch**

A very sensitive switch used to power on and off in many devices. In our scara bot this was used to control the rotation of the arms at each joint.

******

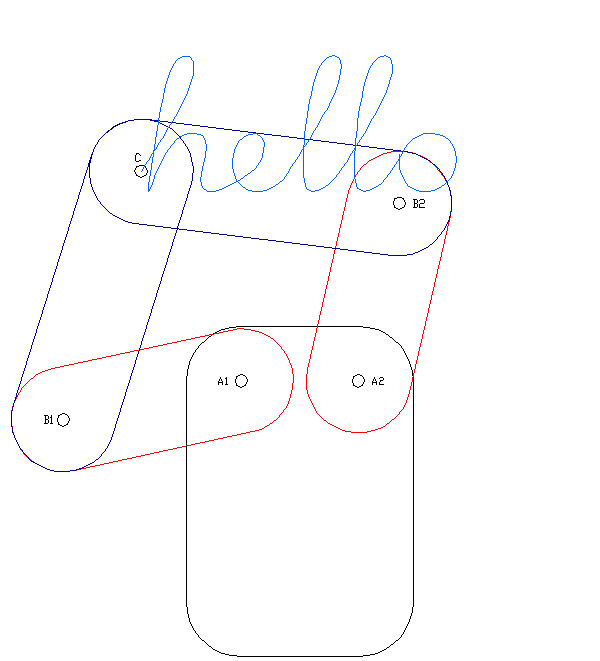
1. **Controller board**

* The controller board, also referred to as the motherboard or main board is the brain of the 3D printer.
* It is the one responsible for the core operation, directing the motion components based on commands sent from a computer and interpreting input from the sensors.
* Controller board is made up of two components, a microcontroller and a circuit board.
* Both works simultaneously to control and distribute power to all other components.
* The controller board’s quality has a major effect on the overall performance of the 3D printer.  
  E.g., Arduino Mega, Smoothie board, SKR Turbo.

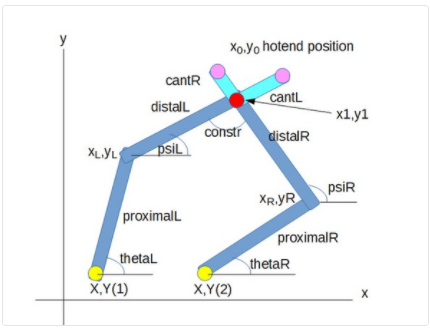
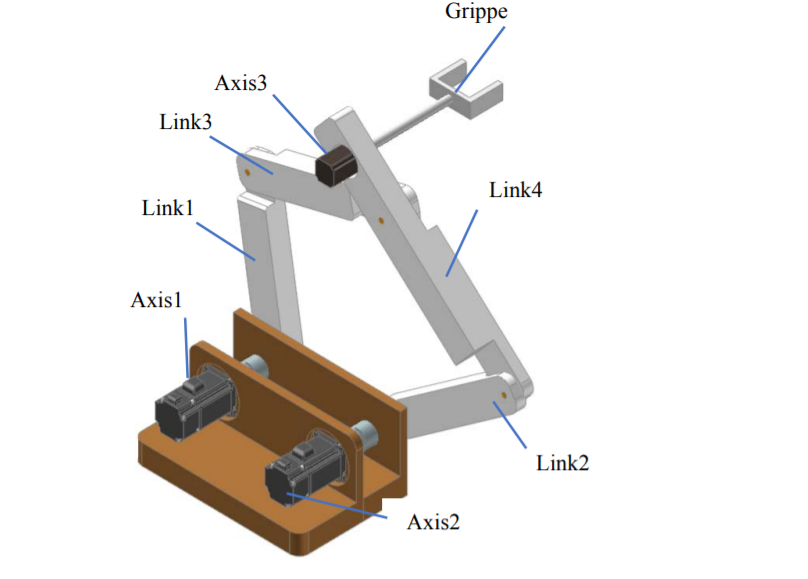
**Mechanism of Parallel arm SCARA bot**

**Working principle of SCARA:**

* arm to extend into confined areas and then retract or "fold up" out of the way

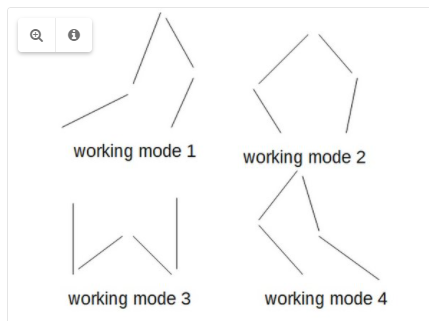


**Construction of parallel arm SCARA bot:**

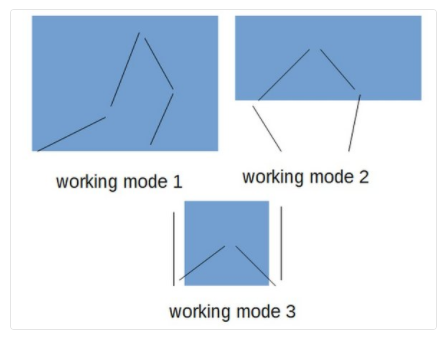
 

* The red dot is the hotend position, the yellow dots represent the actuators/steppers.
* At top are optional cantilevered arms with hotend positions at the end.
* proximalL, distalL are the left arms in mm, proximalR and distalR the right ones. theta and psi left (L) and right (R) are degrees measured from the x axis. The optional cantilever arm of distalL is of length cantL, for distalR it is cantR.
* constr is a constraint angle: if the angle is too flat, the SCARA cannot be moved (singularity). If it is too small, the x0,y0 position becomes undefined.
* x0,y0 is the hotend point, xL,yL and xR,yR the left and right hinges.

These are 4 working modes of a 5 arm SCARA bot. With working mode 1 has the largest printing area:



**Print Area:**

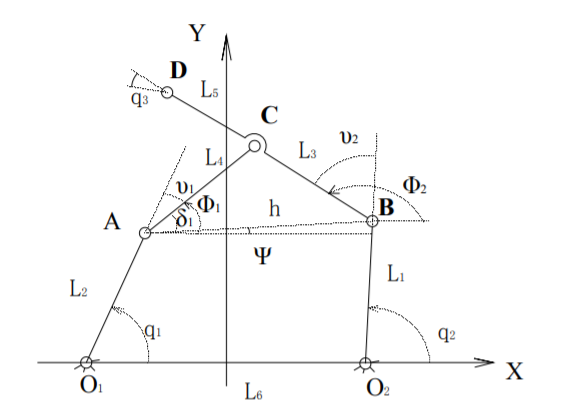


The print area depends on the arms lengths, the distance between the Actuators and the constraints.

**Kinematics of parallel arm SCARA:**

**Forward kinematics:**

In the forward kinematic problem, it is assumed that the joint variables are given and the problem is to find the position and orientation of the end-effector.



Where:

* Oi denotes the fixed base points
* Li denotes the links’ lengths
* qi denotes revolute angle for joints

In forward kinematics q1, q2, q3 are given we need to find, position and orientation of the end-effector G = [XD, YD, α]

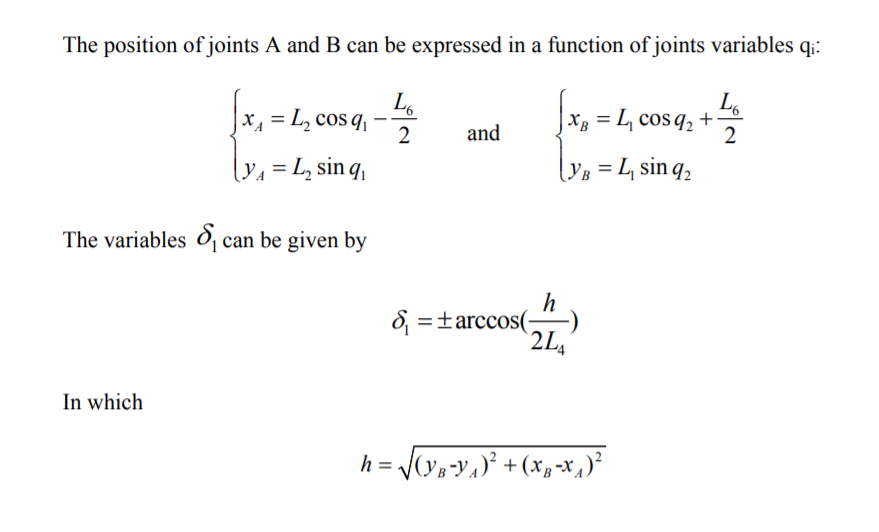
By taking projection we get coordinates of D 

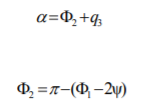
𝜱1 can be obtained by adding 𝝳1 and Ѱ (observe figure)

Ѱ is obtained from triangle.

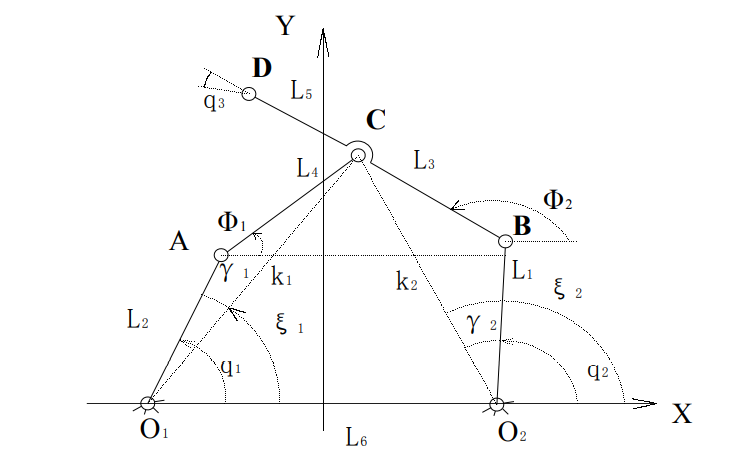
* What is arctan2(x, y)?



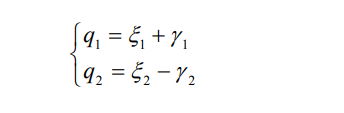


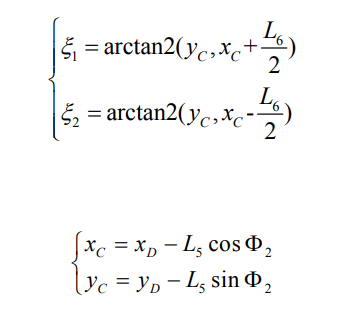
h is obtained using distance formula

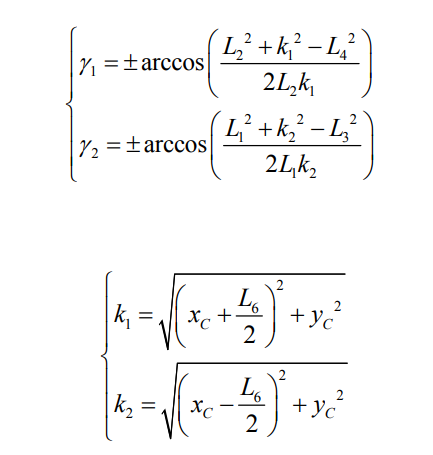
**Inverse Kinematics:**

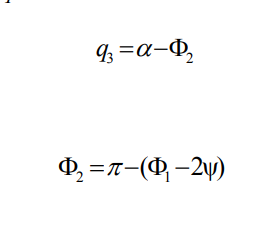
Position and orientation of end-effector (G = [XD, YD, α]) is given, [q1, q2, q3] is need to be found

q1 and q2 is equal to



ε1 and ε2 can be obtained by triangles with hypotenuse O1C and O2C respectively. 

Taking projection of L5 to get coordinates of C, γ1 and γ2 can be obtained by using cosine rule

K1 and K2 is obtained by applying distance formula 

**What is Cooperative 3D Printing?**

Cooperative 3D printing is an emerging technology that aims to provide scalability to 3D printing by enabling thousands of printhead-carrying mobile robots to cooperate on a single printing job and to integrate pre-manufactured components during the 3D printing process. The idea is to have autonomous systems that cooperate with each one understanding what is the overall goal and how each one can contribute.

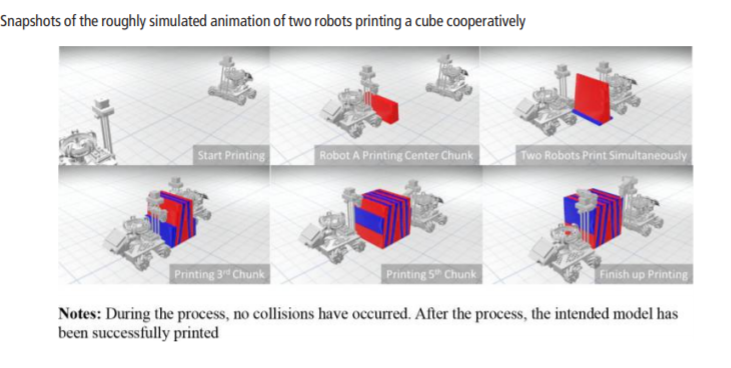
* Consists of multiple 3D Printers that cooperate with each other to print a structure that can be bigger in size than the individual 3-D printers.
* It employs additive manufacturing in its process.
* The entire structure is initially broken down to pieces and the pieces are individually printed by the printers in coordination with each other.
* The advantage is increased printing capacity, which leads to reduced human intervention.
* One way of cooperative 3D printing is to employ the process of *Chunk based slicing*- wherein an object is divided into chunks, and a swarm of printers print different chunks simultaneously.

**CHUNK BASED SLICING**

The chunk-based slicing algorithm is critical to the success of cooperative 3D printing, which may enable an autonomous factory equipped with a swarm of autonomous mobile 3D printers and mobile robots for autonomous manufacturing and assembly.In chunk-based printing, the workpiece is first divided into small chunks and then the chunks are allocated to an army of robots for printing.By enabling multiple mobile 3D printers working together, the printing speed can be significantly increased and the printing capability (for multiple materials and multiple components) can be greatly enhanced.The chunk-based approach keeps the 3D printing local and avoids the large temperature gradient and associated internal stress as the size of the print increases*.*Although the chunk-based printing has demonstrated its capability in speeding up the printing process and scaling up the printing size, the bond strength at chunk joint is still unclear. The lack of this knowledge limits the potential of chunk-based printing. To this end, we assess the tensile strength of chunk-printed parts and compare their strength against those normally printed by traditional layer-based 3D printing. *We first identify the parameters associated with chunk-based printing, such as the chunk slope angle and the chunk overlapping depth, which can directly influence the bond strength. Then, the design of the experiment is performed based on different combinations of these parameters.*

A general strategy that is employed during cooperative 3D printing involves three components:

* Chunker: A CAD model of the print job is input in the chunker. It then splits the print job into *chunks* based on certain criteria.
* Slicer: The chunks of the print job are *sliced*  into multiple layers using a slicer. The slicer also contains commands that are to be followed during the actual printing process- eg: temperature control, tool paths, material extrusion. It also relays information about sequencing the printing of different chunks among multiple robots and enables communication channels among the multiple robots. Thus, *a chunk-based slicer has slightly different functions* when compared to a regular slicer. A regular slicer only generates the tool path.
* Simulator:The simulator analyses the information relayed by the slicer. It enables the visualisation and animation of the print job.



**WHAT IS FIRMWARE?**

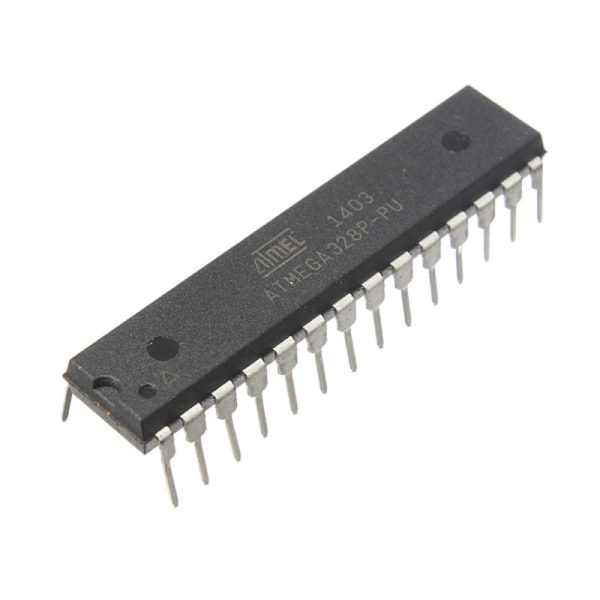
* The firmware is a software installed on the electronics that provides a link between software and hardware.
* It is permanently installed in a machine, device, or microchip, usually by the manufacturer.
* Unlike standard software, firmware is meant to control, operate, or maintain the hardware in the background, and not interact with human users. It is typically used as a low-level software.
* We cannot change a device firmware and can be done only by the manufacturer.
* Since it is planted into the hardware, the firmware is also called as ‘embedded software’.

**3D PRINTER’S FIRMWARE**

* Firstly, the G-code file is a list of commands and coordinates that describes the 3D model to be printed.
* The firmware interprets the G-code files, and performs operations based on known dimensions and heater sensor profiles.
* It is a library of operational protocols that can be tuned by printer vendors and individual innovators alike, to improve or completely alter performance characteristics.
* So it is the use of the machine specific firmware in combination with the control electronics component, and the job specific G-code instructions, that produces the printed object.

**MICROCONTROLLER (MCU)**

* The microcontroller is the brains of the 3D printer.
* A microcontroller is basically a fully functionalized computing system, or a computer. The difference between the two is that microcontroller does not dedicate to general purpose computational tasks, they are designed for embedded applications, providing control logic for modern electronic devices
* It allows you to manage four bipolar stepper motors to drive the mechanics of a 3D printer (X, Y, Z and a fourth motor for the extruder), the heater of the extruder, the heated plate and a fan.
* It communicates with the stepper motor drivers to tell them when to stop and go.



For example :

G1 X50 Y20 E15

Here G1 instructs the microcontroller to prepare for a linear movement. Then the new coordinates, which are X = 50, Y = 20, and E = 15, are read in by the firmware. For 3D printers, the E stands for the extrusion drive. When an E is present in a G code coordinate then either the filament is being deposited or retracted.

Now the firmware performs some basic calculations to determine how far off each carriage is from the required positions then the microcontroller will send step pulses to the stepper drivers, which will move the appropriate stepper motors.The above sequence of events relies on the assumption that the the firmware understands the exact configuration of the 3D printer down to the transmission specifications for each axis and extrusion drives.

**CONFIGURATION VARIABLES**

The appropriately named configuration variables are where our printer’s specific information is stored. These variables are typically located in an accessible file known as a configuration file. Now the microcontroller of a 3D printer has a long list of G-codes that needs to be turned into a physical 3D object, so it needs the information stored in the firmware to make that happen. So the information we need are :

* Number of extruders – How many extruders does the 3D printer have?
* Heated bed – Is there a heated bed?
* Location of limit switches – Are the limit switches at the minimum or maximum of their respective linear guides?
* Z probe – Is there a Z-probe? If not, then how far away is the extruder from the build plate?
* Stepper motor rotation to linear motion – For each axis, how far does the carriage move per rotation of the stepper motor?
* Build Volume – What is the maximum travel for each axis?
* Direction – Does clockwise or counterclockwise rotation of each stepper motor result in positive movement?
* Filament extrusion – How much filament is extruded per rotation of the extrusion drive stepper?
* Speed – What is the maximum speed each motor is capable of?

Across different firmwares these configuration variables have slightly different names, inputs and locations.

step pulse to linear distance conversion :

The microcontroller sends step pulses to the drivers in order to move the motors. For the microcontroller to properly move the carriages to new coordinates it must know precisely how far the carriages move per pulse that it sends. This step pulse per some distance will depend on how the transmission is set up and therefore each axis, including the extrusion drive, could have a different step pulse to linear distance conversion.

This is a snippet of code used in ‘RepRapFirmware’ to set these step pulse to linear distances variables for each axis and extrusion drive:

M92 X80 Y80 Z1200 E410

The M92 command tells the RepRapFirmware to set how many step pulses it takes to cover 1 mm of linear distance for the following axes.

* The X carriage moves 1 mm every 80 step pulses.
* The Y carriage moves 1 mm every 80 step pulses.
* The Z axis moves 1 mm every 1200 pulses.
* The E parameter again refers to the extruder and here it takes 410 step pulses to extrude 1 mm filament.

There are many different firmwares and some of the commonly used are Marlin, Smoothie, and RepRapFirmware.

**SMOOTHIE FIRMWARE**

In its most basic form, the job of Smoothie is to receive G-code commands, and translate those into actual movement for the printer.

This is done in several steps :

* The SerialConsole module reads the serial port ( UART or USB ), and when a line is recognized, it follows it to all modules that asked for it by triggering the ‘on\_console\_line\_received event’.
* The GcodeDispatch module is one of those that is registered for the ‘on\_console\_line\_received event’, and is thus called every time the SerialConsole module triggers it. It takes the new line, and if it recognizes a G-code command, transforms it into a new Gcode object, and triggers the ‘on\_gcode\_received event’.The G Code object is just a wrapper around the actual string, it just provides helper function to retrieve values from that string.
* The Robot module listens to this event, and is triggered. It uses math to cut the requested move into line segments, and passes those to the Planner module. There they are transformed into Block objects, containing speed, direction and acceleration information. The acceleration profile for the Planner's queue ( list of upcoming Blocks ) is re-computed to take the new Block into account, and finally that Block is added to the queue.
* The Stepper module itself enters the game whenever there is a Block in the Planner's queue and is composed of two loops :

1. The stepping loop, which pops new blocks if necessary, and actually sends the step and direction command to the stepper motor drivers to move the motors.
2. The acceleration loop, which updates the stepping loop's speed depending on the acceleration profile for this block.

**REPRAP FIRMWARE**

RepRapFirmware is a comprehensive motion control firmware intended primarily for controlling 3D printers, but with applications in laser engraving/cutting and CNC too. Unlike most other 3D printer firmwares, it is targeted only at modern 32-bit processors, not outdated 8-bit processors with limited CPU power. So it is designed to make good use of the power of modern inexpensive ARM processors to implement advanced features. It is configured with human editable files located on an SD-card plugged into the printer electronics. As such, there is no need for ordinary users to compile the software nor install any development tools.

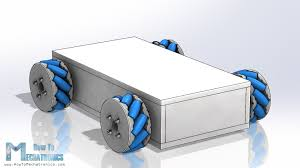
The software can receive G-Code from the USB port, the serial port, the SD card, the Ethernet or WiFi interface via http, and the Ethernet interface via Telnet. Although it can run from an interface connected via USB, most users prefer to operate from a web interface, with the board connected to an Ethernet or WiFi network. Files to be printed are uploaded to the SD card via the web interface. The RepRap Firmware philosophy is that every operation is done with G-code, including configuration.The configuration file is read on SD card at start-up.Any G-code or macro could be sent to the board while operating the printer, allowing instant feedback for any configuration modification. As interactive modifications are lost at next board start-up, the successfully tested G-Codes shall be manually introduced in the configuration file, which can be edited directly in the web interface. This interactive configuration makes printer commissioning and tuning easier than with most other firmwares.

**WHEELS FOR A MOBILE SCARA 3D PRINTER**

**MECANUM WHEELS**



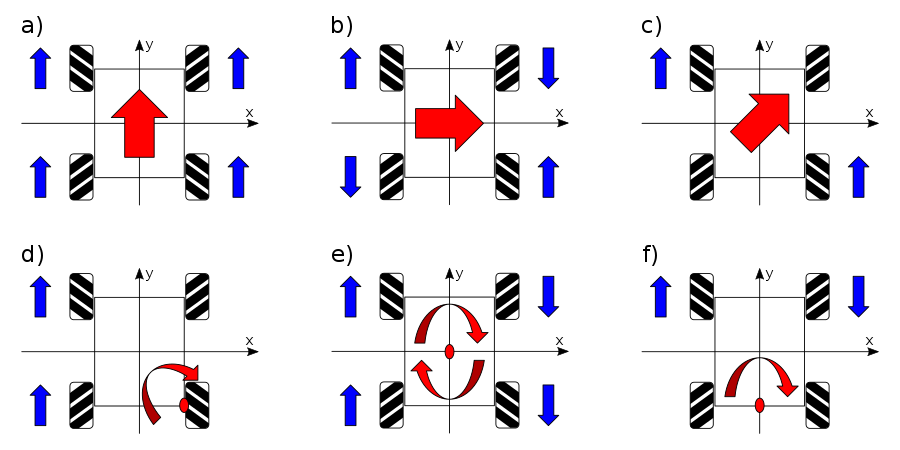
The Mecanum wheel is based on a tireless wheel, with a series of rubberized external rollers obliquely attached to the whole circumference of its rim. These rollers typically each have an axis of rotation at 45° to the wheel plane and at 45° to the axle line. Each Mecanum wheel is an independent non-steering drive wheel with its own powertrain, and when spinning generates a propelling force perpendicular to the roller axle, which can be vectored into a longitudinal and a transverse component in relation to the vehicle.



This is a typical four mecanum wheel model

This will be the most efficient choice for a 3D printer due to its multi directional movement unlike the typical powertrain with steering.

STEPPER MOTORS are used for designing 3D printers for more accurate positioning.

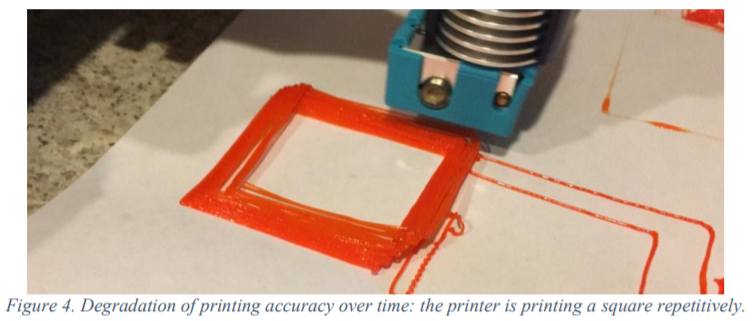


Explanation of the movement corresponding to each configuration

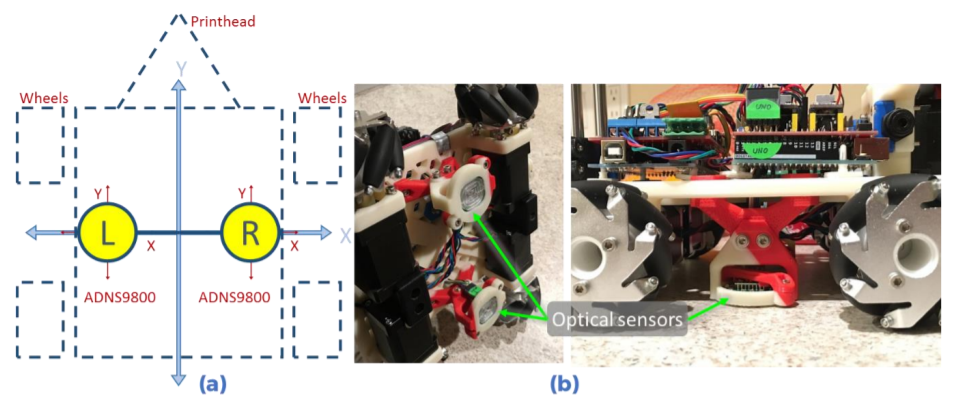
<https://www.youtube.com/watch?v=Xrc0l4TDnyw>

**POSITIONING**

Although we predict slipping will not be a significant issue if we cap the maximum acceleration of the robot, it still happens occasionally due to imperfect contact between the wheels and the floor and imperfect alignment of the wheels, which leads to degradation of printing accuracy over time as shown



* Among various position sensors, an optical mouse sensor satisfies our requirements. It takes a series of pictures at very fast rate and determine the relative movement by comparing the pictures using an optical flow algorithm
* ***laser mouse sensor (ADNS-9800 sensor) performs better and its positioning accuracy is within 5%.***
* In addition to the XY movement, the mobile printer is subjected to one additional degree of freedom (DOF) – rotation about the Z axis due to inaccuracy in the wheels. The orientation of the robot must be corrected to perform quality printing over time.
* However, the IMU sensors are not accurate enough to report the correct orientation of the robot due to the small and slow rotation of the robot during printing (the robot usually only rotates a few degrees over 20 minutes).
* a solution by using two optical mouse sensors as shown



**Role of Firmware**

The firmware specific for our mobile variant must feed the normal G-code created by slicer into commands for the motion of wheels which is the sole idea of the project

**Steps involved in 3D Printing**

1. Make a CAD model.
2. Use a slicer to slice it into layers and generate a G-code.
3. Interpret this G-code and give instructions to the 3D Printer what to do.