Dragon-arm one gen1 User Manual

Product: Dragon-arm one gen1 **Maker:** SpaceDragon robotics **Revision:** A

Robot type	Controller	Use class
5-axis stepper-driven articulated arm	Arduino UNO R4 / UNO Q + 6 × TMC2209	Hobby / prototyping

1. Product Overview

Dragon-arm one gen1 is a 5DOF 3D-printable desktop robotic arm project, Every axis is driven by a NEMA 17 stepper motor. Axes 1–4 use a two-stage 9:1 belt reduction, and the shoulder (Axis 2) is driven by two motors working together. Axis 5 rotates the tool directly, without reduction. Each axis can carry an MT6701 absolute magnetic encoder, so the arm always knows its joint angles.

The arm is controlled by an Arduino UNO R4 or UNO Q. Six TMC2209 stepper drivers receive STEP/DIR signals from the Arduino, plus one shared enable (EN) line and one shared UART line used to set the step mode. The supported way of programming the arm is the SpaceDragon programmer app. Users may also write their own Arduino code.

Dragon-arm one gen1 is a hobby and prototyping robot, distributed as print-it-yourself files under the Cults Private Use (PU) license. It is not a certified industrial or safety-rated machine.

1. Product Overview **1**
2. Key Features **3**
3. Technical Specifications **3**
3.1 General Specifications 3
3.2 Axis Specifications 4
3.3 Kinematics and Workspace 4
3.4 Material and Print Information 5
3.5 Required Components 5
4. Mechanical Interfaces **6**
4.1 Base Mounting 6
4.2 Tool Interface 7
5. Electrical Interfaces **7**
5.1 Control Electronics 7
5.2 Power Supply 8
5.3 Arduino Pin Assignment 8
5.4 Driver Board 8
5.5 Wiring Diagram 10
6. Control and Software Interface **11**
6.1 SpaceDragon Programmer App 11
6.2 Control Architecture 11
6.3 Driver Configuration 11
6.4 Encoders and Direction Check 11
7. Startup and Operating Notes **12**
7.1 Startup 12
7.2 Shutdown 12
7.3 Operating Notes 12
8. Maintenance **12**
9. Troubleshooting **12**
10. Environmental Information **12**
10.1 Environmental Specifications 12
11. Safety Information **13**
11.1 Intended Use 13
11.2 Restrictions 13
11.3 Handling Warnings 13
12. Product Images **13**
13. Support and License **14**
14. Revision History **14**

2. Key Features

• 5 stepper-driven axes: base, shoulder, 2 arm segments, tool roll	• Maximum reach 540 mm
• 9:1 two-stage belt reduction on Axes 1–4	• Dual-motor shoulder (Axis 2)
• Payload 300 g in any pose, up to 500 g when not fully extended	• Optional MT6701 absolute encoder on every axis
• Arduino UNO R4 / UNO Q with 6 × TMC2209 drivers	• 12–36 V DC supply (24 V recommended), power switch in base
• Fully 3D-printable structure (~2 kg filament, PETG/PCTG)	• Programmed with the SpaceDragon programmer app



Figure 1. Dragon-arm one gen1, arm extended forward (side view)

3. Technical Specifications

3.1 General Specifications

Parameter	Value
Robot type	Stepper-driven articulated arm, belt reduction
Degrees of freedom	5 (base yaw, shoulder, Segment 2, Segment 3, tool roll). End effector not included
Weight	4.15 kg
Base footprint	≈ 293 × 190 mm including mounting lugs
Axis 2 (shoulder) height	125 mm above mounting surface
Maximum reach	540 mm from Axis 2 to the tool mount
Maximum Height, arm vertical	≈ 665 mm to the tool mount
Rated payload	300 g in any pose and condition; up to 500 g stable payload when the arm is not fully extended
Accuracy	±3 mm
Controller	Arduino UNO R4 or Arduino UNO Q

Parameter	Value
Stepper drivers	6 × TMC2209 StepStick modules (or compatible)
Supply	12–36 V DC, ≥ 9 A (24 V recommended; tested at 24 V)
Position feedback	5 × MT6701 absolute magnetic encoders, analog output (optional; required by the SpaceDragon programmer app)
Base mounting	4 × Ø8.2 mm through-holes (see section 4.1)

Performance note: Accuracy and repeatability depend strongly on the user's build (print quality, belt tension, driver current) and on the control code.

3.2 Axis Specifications

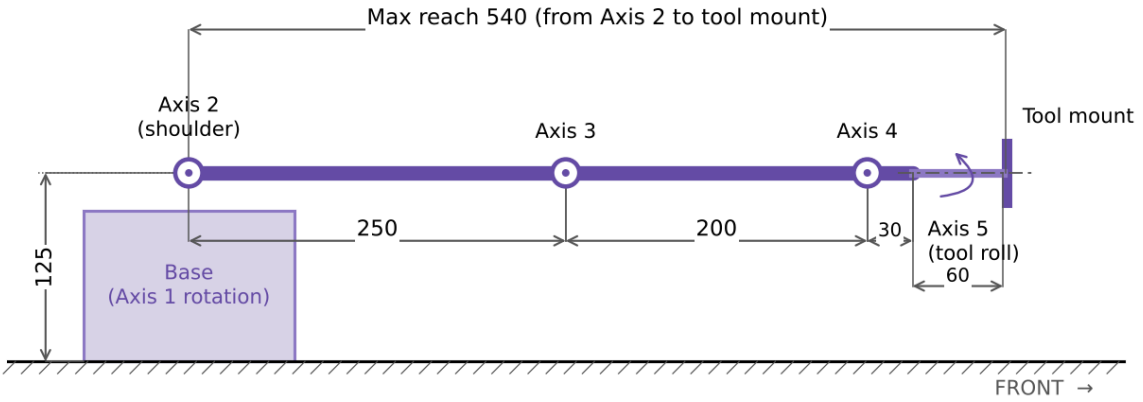
Axis	Function	Motor	Reduction	Range	0° position
Axis 1	Base rotation (yaw)	NEMA 17 42Ncm	9:1	−150° to +150°	Robot front
Axis 2	Shoulder (Segment 1)	2 × NEMA 17 59Ncm	9:1	0° to 135°	Maximum back position
Axis 3	Segment 2	NEMA 17 59Ncm	9:1	0° to 240°	Maximum back position
Axis 4	Segment 3	NEMA 17 42Ncm	9:1	0° to 180°	Maximum back position
Axis 5	Tool rotation (roll)	NEMA 17 12Ncm	1:1 (direct)	0° to 360°	Pointing back, toward the robot

Parameter	Value
Drive type (Axes 1–4)	Stepper motor + two-stage HTD 3M belt reduction, 10 T → 30 T per stage (3:1 × 3:1 = 9:1)
Drive type (Axis 5)	Direct drive, no reduction
Recommended step mode	Full step, set over the shared UART line (section 6.3)
Joint resolution at full step	Axes 1–4: 0.2° per step (5 steps/°). Axis 5: 1.8° per step (calculated for 200-step motors)
Joint limits	Physical (mechanical) limits only, as listed above. There is no rule coupling two axes
Zero positions	Defaults as listed above; easily changed in the SpaceDragon programmer app
Speed and acceleration	Set in the user code / programmer app
Position feedback	MT6701 absolute encoder per axis (optional). Read by the Arduino analog inputs

Developer Note: Joint resolution follows from motor steps × reduction. A 1.8° motor has 200 full steps per revolution; with 9:1 reduction that is 1800 steps per joint revolution, i.e. 0.2° per step. If you change the step mode or use a 0.9° motor, recalculate steps per degree in programmer app settings.

3.3 Kinematics and Workspace

Parameter	Value
Axis 2 height	125 mm above mounting surface
Segment 1 (Axis 2 → Axis 3)	250 mm
Segment 2 (Axis 3 → Axis 4)	200 mm
Axis 4 → Axis 5	30 mm
Axis 5 → tool mount	60 mm
Maximum reach	540 mm from Axis 2 to the tool mount (all segments collinear)
Reach to tool point	Maximum reach + length of the mounted tool



Arm shown fully extended forward (illustrative pose). Dimensions in mm.

Figure 2. Kinematic layout, side view (dimensions in mm)

3.4 Material and Print Information

Parameter	Value
Recommended material	PLA-CF or PLA Plus
Filament required	~2 kg
Printer	Minimum 200 × 200 mm bed. Below 300 × 300 mm the base plate is printed in two parts
Standard parts	0.2 mm layer, 2 walls, 15% infill, 3 top / 3 bottom layers
Gears	5 walls, > 30% gyroid infill, 5 top / 5 bottom layers
Wall generator	Classic

3.5 Components

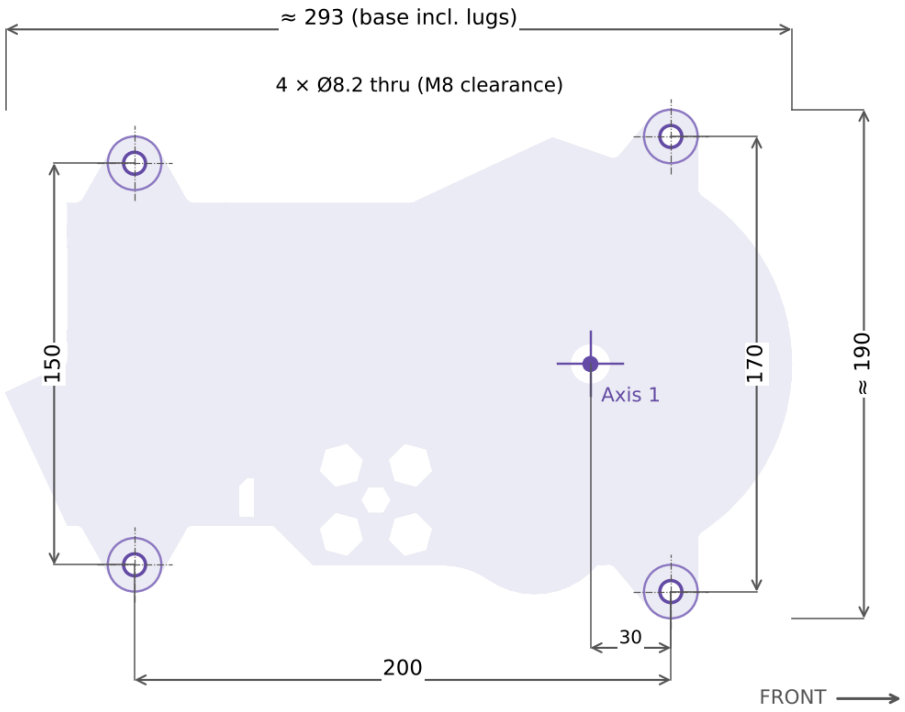
Parameter	Value
Controller	1 × Arduino UNO R4 or UNO Q (Arduino screw shield recommended)
Stepper drivers	6 × TMC2209 StepStick (or compatible)
Stepper motors (NEMA 17)	2 × 38 mm (≥ 42 N·cm), 3 × 48 mm (≥ 59 N·cm), 1 × 26 mm (≥ 12 N·cm)
Encoders (optional)	5 × MT6701, calibrated with the AA encoder calibrator
Bearings	22 × 15 × 24 × 5 mm
Belts	HTD 3M, 4–10 mm wide: 7 × 225 mm, 1 × 348 mm. GT2, 2–6 mm wide: 1 × 150 mm
Power supply	12–36 V DC, ≥ 9 A (24 V recommended). Internal bay fits 50 × 205 × 23 mm
Power switch	1 × power entry + switch, 12 × 19 mm cut-out (in the base)
Fan	1 × 60 × 60 × 10 mm, rated for the supply voltage
Threaded inserts	> 50 × M3
Screws	> 50 × M3 × 8 mm countersunk
Other	Cables, Dupont jumpers, cable sleeves (optional), ~2 kg filament

Note: The full part list, tools and step-by-step build procedure are given in the Dragon-arm one gen1 Assembly Instruction.

4. Mechanical Interfaces

4.1 Base Mounting

Parameter	Value
Mounting holes	4 × Ø8.2 mm M8 through-holes in the base lugs
Front hole pair	30 mm in front of Axis 1, 170 mm apart
Rear hole pair	170 mm behind Axis 1, 150 mm apart
Front-to-rear hole spacing	200 mm
Base footprint	≈ 293 × 190 mm including lugs



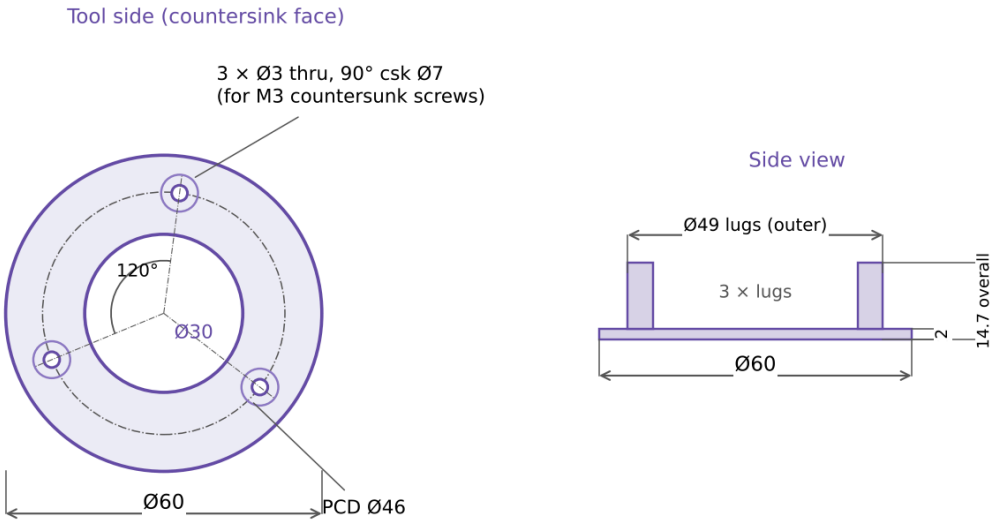
Top view of base. Hole positions measured from Axis 1. Dimensions in mm, taken from the base CAD model.

Figure 3. Base mounting pattern, top view (dimensions in mm)

Important: Bolt the base to a rigid surface before working with payloads or fast moves. A 500 g payload far from the base creates a large tipping moment, and fast moves add inertial loads.

4.2 Tool Interface

Parameter	Value
Interface	Claws tool-mounting system on Axis 5
Tool mount plate	Ø60 mm, 2 mm flange, 14.7 mm overall height, Ø30 mm centre bore
Tool fixing holes	3 × Ø3 mm through, 90° countersink Ø7 mm, on a Ø46 mm pitch circle, 120° apart, for M3 countersunk screws
Tool / end effector	Not included. Reach and payload at the tool point depend on the tool used



Tool mount plate (general parts/tool mount). Dimensions in mm, taken from the CAD model.

Figure 4. Tool mount plate (dimensions in mm)

Pro Tip: To build your own tool, design it with three M3 threaded inserts on a Ø46 mm pitch circle at 120°, screw the tool mount plate onto it with M3 countersunk screws, then fit the plate to the claws on Axis 5. Leave the Ø30 mm centre free if cables need to pass through.

5. Electrical Interfaces

Dragon-arm one gen1 uses an Arduino UNO R4 or UNO Q, six TMC2209 drivers on a StepStick carrier, and an internal DC power supply. Motor power and logic power are separate; all grounds are common.

5.1 Control Electronics

Parameter	Value
Controller	Arduino UNO R4 or UNO Q
Stepper drivers	6 × TMC2209 (Axis 2 uses two drivers on the same STEP/DIR signals)
Driver control	STEP/DIR per axis, one shared EN line (active LOW), one shared write-only UART line
Feedback	MT6701 encoders, analog output to Arduino A1–A5
Controller power	Arduino USB port
Driver logic and encoder power	5 V and GND from the Arduino
Motor power	DC supply V+ to the driver board VM input
Cooling	60 mm fan on the supply output, always on while main power is on
Power switch	In the base, switching the supply input (AC line)
Emergency stop	Not included

5.2 Power Supply

Parameter	Value
Supply voltage	12–36 V DC; 24 V recommended (built and tested at 24 V)
Upper voltage limit	Set by the drivers fitted. TMC2209 modules: 29 V absolute maximum, so stay at 24 V or below. Higher voltages only with drivers rated for them
Supply current	≥ 9 A
Supply mounting	Internal bay 50 × 205 × 23 mm, or an external supply
Common ground	Required between Arduino, driver board, encoders and the supply

Parameter	Value
Reverse-polarity protection	None on the driver board. Fuse and other protection optional but recommended

Important: The power switch and supply input carry mains voltage. Wire them with insulated, strain-relieved connections and close the base before connecting to mains. If you are not qualified to work with mains wiring, use an external, enclosed power supply.

5.3 Arduino Pin Assignment

Arduino pin	Signal	Connected to
D2 / D3	STEP / DIR	Axis 1 driver
D4 / D5	STEP / DIR	Axis 2 drivers (both motors)
D6 / D7	STEP / DIR	Axis 3 driver
D8 / D9	STEP / DIR	Axis 4 driver
D10 / D11	STEP / DIR	Axis 5 driver
D12	EN (active LOW)	All six drivers
A0	UART TX (write-only)	All six drivers (PDN_UART)
A1 – A5	Analog input	MT6701 encoders, Axis 1 – Axis 5
5V / GND	Logic supply	Driver board logic, encoders
D0 / D1	USB serial	Not connected (used by USB)
D13	—	Not used

Note: Avoid D13 for EN: it blinks while the bootloader runs, which would briefly enable the drivers.

5.4 Driver Board

The TMC Driver Board v2 carries six StepStick drivers. Channel n is always driver An, motor connector Jn and pads STEPn / DIRn. All connections are solder-wire pads except the motor connectors.

Pad / connector	Connect to	Notes
24V / GND	Power supply + / −	Thick wire (0.5 mm² or more)
5V	Arduino 5V	Drivers' logic supply
Logic GND	Arduino GND	Always connect, even with a USB-powered Arduino
STEP1–6 / DIR1–6	Arduino digital pins	One pair per channel (section 5.3)
EN	Arduino D12	Shared, active LOW, 10 kΩ pull-up on the board
UART	Arduino A0	Shared, write-only (R2 = 1 kΩ)
J1–J6	Stepper motors	JST-XH 4-pin, order 1B, 1A, 2A, 2B

Driver setup checklist
• Orientation: the module's VM pin goes next to that driver's capacitor. A driver inserted backwards is destroyed at power-up.
• UART: for full step, the module must have UART on header pin 4 (the "MS3" position). Some modules need a solder bridge.
• Current: set each module's Vref before connecting a motor, using that module vendor's formula. Aim for 70–85% of the motor's rated current.
• Cooling: fit a radiator on every driver. The base fan blows across the drivers.

5.5 Wiring Diagram

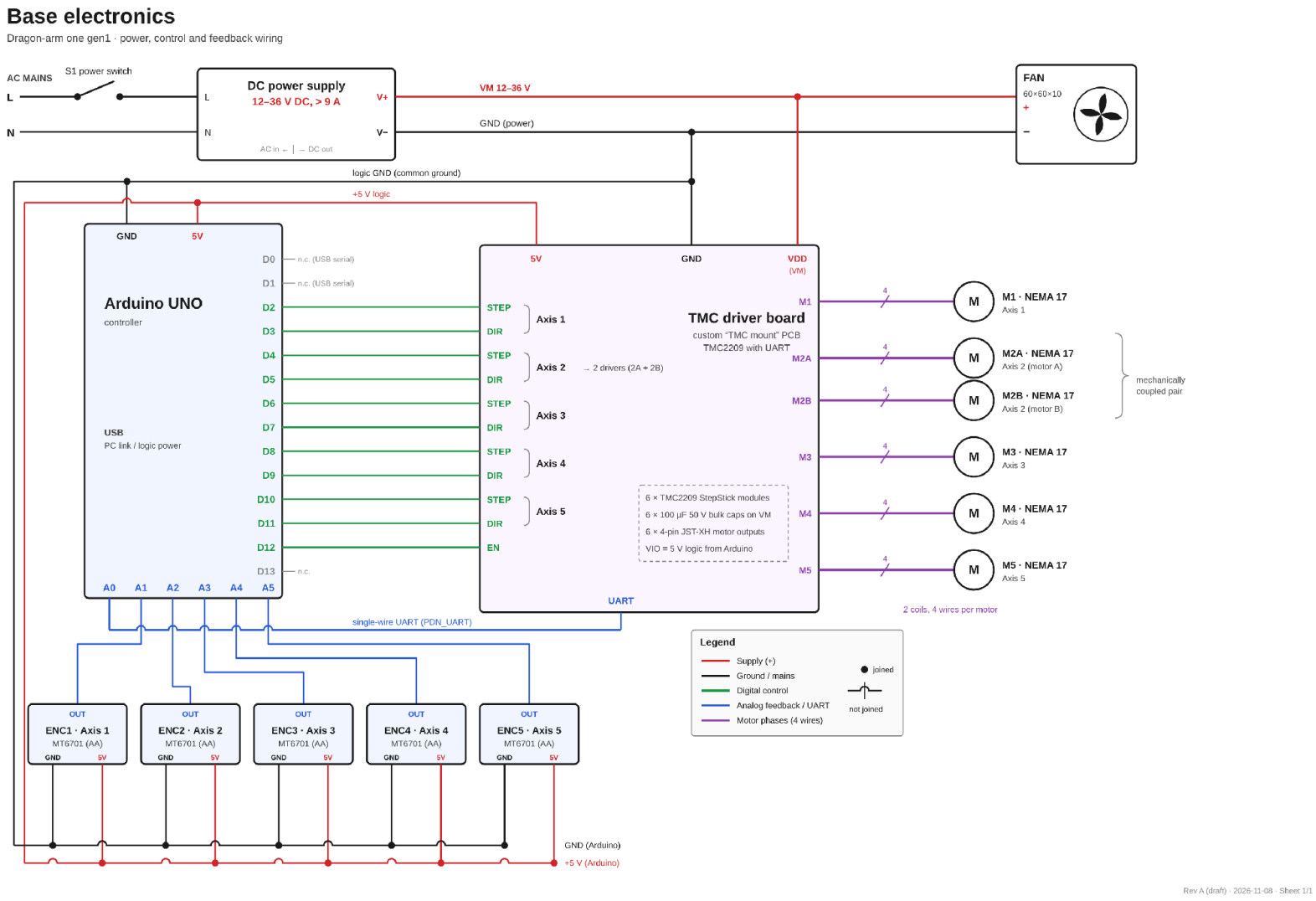


Figure 5. Base electronics: power, control and feedback wiring

6. Control and Software Interface

6.1 SpaceDragon Programmer App

The supported way of programming Dragon-arm one gen1 is the SpaceDragon programmer app. Movements are planned with command blocks or by animating a 3D model of the robot, and the app generates the Arduino code that runs on the arm. The app is available at:

https://spacedragon.online/spacedragon-app/

Parameter	Value
Motion planning	Command blocks and 3D-model animation
Output	Arduino code, uploaded to the controller over USB
Encoders	Required for use with the app
Configurable	Axis zero positions, speed and acceleration, encoder offsets, direction check mode
Diagnostics	Code option that prints encoder angles to the serial monitor

6.2 Control Architecture

Parameter	Value
Typical architecture	SpaceDragon app or user code → Arduino → TMC2209 drivers (STEP/DIR) → stepper motors
Feedback path	MT6701 encoders → Arduino analog inputs A1–A5
Command level	Joint (axis) angles
Custom control	Any control method may be implemented by the user in Arduino code
ROS / ROS 2	Not officially supported

6.3 Driver Configuration

Full step is recommended. The MS pins of the TMC2209 cannot select a full step, so the step mode is set over the shared UART line. The line is write-only: the same settings go to all six drivers at once and nothing is read back.

Register	Value	Effect
GCONF (0x00)	0x000001C1	Current from the Vref trimmer, UART enabled, step size from register
CHOPCONF (0x6C)	0x18000053	Full step (MRES = 8), interpolation on

Developer Note: The drivers forget these settings when motor power is removed. Send them at every boot, and again whenever main power was switched off while the Arduino stayed on (resetting the Arduino does this). Without UART configuration a TMC2209 runs at its power-up step size (1/32 step on this board), so steps per degree change by a factor of 32.

The drivers stay disabled until the code pulls EN LOW. A 10 kΩ pull-up on the board holds all drivers off while the Arduino resets or is unplugged.

6.4 Encoders and Direction Check

The MT6701 encoders are absolute, so the arm always knows its joint angles and no homing move is needed. Before closed-loop motion the code only has to confirm the direction (DIR) of each motor. This can be done:

- before the program starts (default),
- before each axis moves for the first time, or
- during arm movement.

Each encoder is calibrated with the AA encoder calibrator during assembly. A remaining constant offset, which comes from the encoder board itself, can be calibrated in the SpaceDragon programmer app.

7. Startup and Operating Notes

7.1 Startup

Sequence	Steps	Use when
A. Arduino first	Connect the Arduino over USB, double-click its reset button to put it in standby, then switch on main power	Uploading new code
B. Main power first	Switch on main power, then connect / power the Arduino	Running the code already on the Arduino

7.2 Shutdown

No shutdown sequence is required. The EN pull-up keeps the drivers disabled whenever the Arduino is not driving them: main power without the Arduino only runs the fan, and the Arduino without main power can only read the encoders.

7.3 Operating Notes

- The arm can move as soon as the code pulls EN LOW. Keep clear of the arm when starting a program.
- Without motor power (or with EN HIGH) the arm drops slightly.
- Keep every command inside the axis ranges in section 3.2. There are only mechanical limits.
- No emergency stop is included. To stop the arm, switch off the power switch in the base.
- Never plug or unplug a motor while main power is on. The voltage spike destroys the driver.
- Re-send the driver configuration (reset the Arduino) after main power has been cycled.

8. Maintenance

No scheduled maintenance is required. If the arm's behaviour changes, use the troubleshooting table in section 9.

9. Troubleshooting

Symptom	Likely cause	Solution
Rough motion, binding or skipped steps in one axis	Print defects in the transmission (pulleys, gears, frame)	Inspect and reprint the affected parts. See the Assembly Instruction
Weak motors, missed steps under load	Driver current limit set too low	Set Vref according to the driver datasheet (70–85% of motor rated current)
Random stops, glitches, lost encoder readings	Loose Dupont connections at the Arduino or encoders	Use an Arduino screw shield; check and secure every Dupont connection
Strange moves in closed-loop code	Faulty encoder	Print encoder angles to the serial monitor (app code option or simple own code) and check each axis
Constant position offset on an axis	Encoder board offset remaining after calibration	Calibrate the encoder offset in the SpaceDragon programmer app
Nothing moves, motors hold no torque	EN not pulled LOW, or main power off	Check the EN line (D12) and the power switch
A motor turns the wrong way	Coil pair order reversed	Swap one coil pair at the motor connector, or invert DIR in code
Moves are 32× too short after a power cycle	UART step-mode configuration lost	Reset the Arduino so the driver configuration is sent again

10. Environmental Information

10.1 Environmental Specifications

Parameter	Value
Intended environment	Indoor, dry, desktop use
Operating temperature range	−25 °C to +40 °C
Formal ingress protection rating	None

Parameter	Value
Water exposure suitability	Not intended for contact with water

Note: Printed plastic parts have limited heat resistance. Keep the arm away from heat sources and direct sunlight.

11. Safety Information

11.1 Intended Use

Dragon-arm one gen1 is intended only for hobby projects and prototyping.

11.2 Restrictions

- Do not use Dragon-arm one gen1 with medical devices.
- Do not use Dragon-arm one gen1 with human-interactive safety-relevant machines.
- Do not use Dragon-arm one gen1 with animal-interactive systems.
- Do not use Dragon-arm one gen1 in life-critical or safety-critical applications.
- Do not use Dragon-arm one gen1 exposed to water.

11.3 Handling Warnings

- Do not touch moving parts during operation. Belts, pulleys and joints create pinch points.
- Keep fingers, tools and loose materials out of the arm's workspace while main power is on.
- The arm moves without warning when the code enables the drivers or sends a command.
- The base contains mains wiring. Disconnect from mains before opening the base.
- Verify wiring and supply polarity before power-up. The driver board has no reverse-polarity protection.
- Secure the base before operating with a payload.
- The user is fully responsible for safe assembly, programming and operation.

12. Project Images

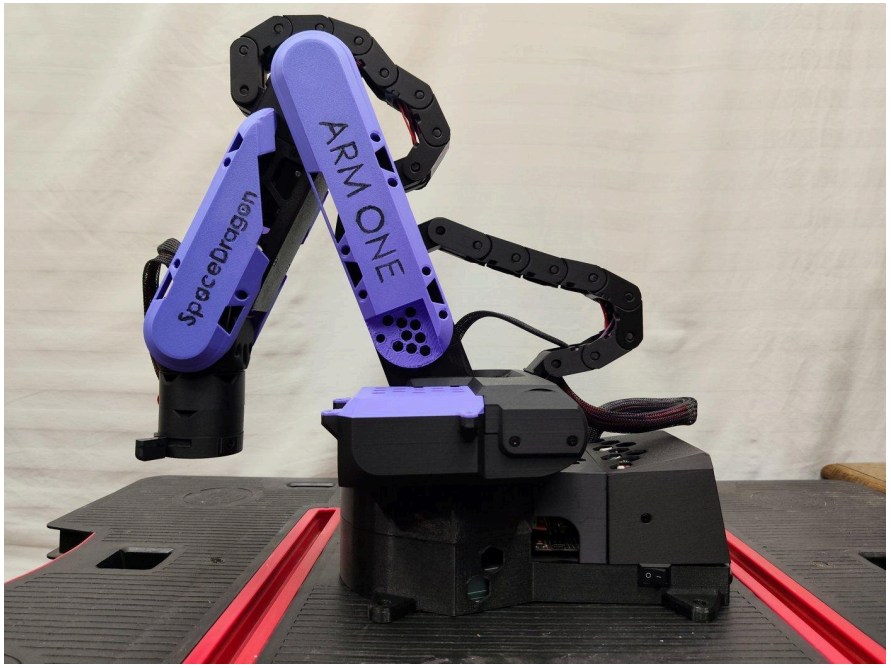


Figure 6. Arm raised, segments folded (side view)

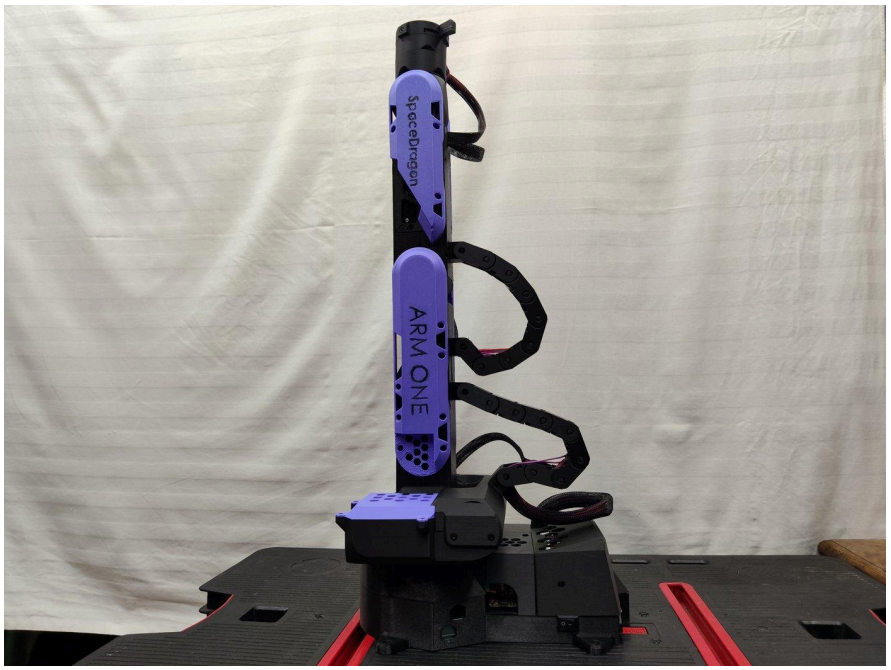


Figure 7. Arm fully vertical (side view)

13. Support and License

Parameter	Value
License	Cults Private Use (PU)
Programmer app	https://spacedragon.online/spacedragon-app/
Community	SpaceDragon community Discord: https://discord.gg/pbjFXMpWYx
Video	SpaceDragon robotics on YouTube: https://www.youtube.com/@SpaceDragon-robotics
Website	https://spacedragon.online

14. Revision History

Parameter	Value
A (2026-11-10)	Initial release