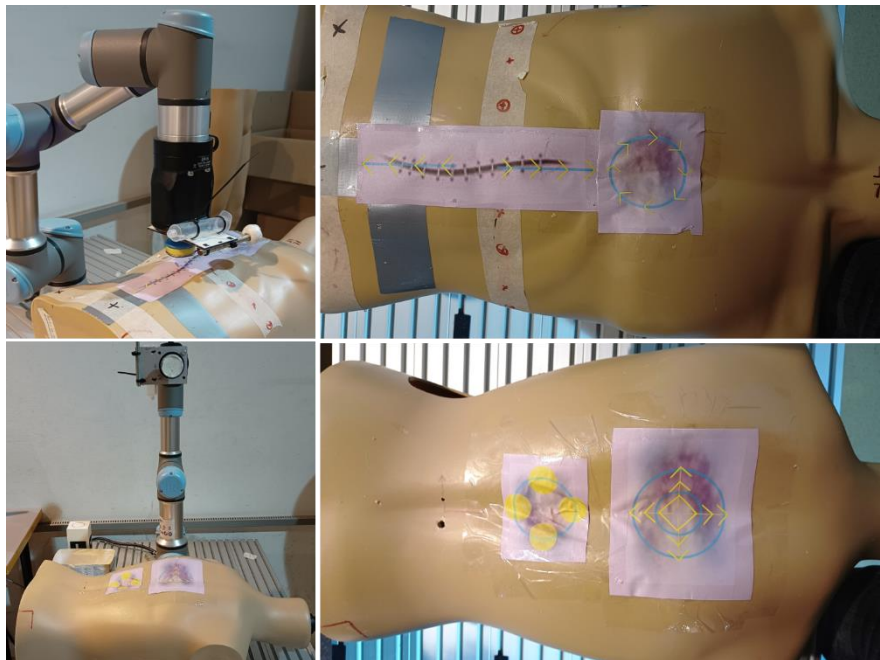# *Wound Cleaner Robot*

LAB REPORT

Computed Aided Surgery and Medical Robotic

Agustin CARTAYA

Micaela RIVAS

Hsham NGIM

Date: 08/01/2024

1. Project Explanation

Objective of the project

Over the last few decades, robots have been playing an increasingly important role in the medical domain, assisting experts in performing a wide range of operations [1,2,3,4,5]. The objective of this project is to develop a robotic system, named "WoundCareBot", capable of performing wound cleaning in an automated and precise manner.

Wound Cleaning

A wound is an injury that breaks the skin or other body tissues. There are different types of wounds, and their treatment depends on their nature [6]. Cleaning the wound is an essential step, as it helps remove contaminants, bacteria, dead tissue, and excess fluid from both the wound and the surrounding skin. This process accelerates healing and reduces the risk of infection [7]. Typically, this is done using a saline or antiseptic solution applied with gauze or cloth [8].

For surgical wounds, the best way to clean is a linear movement starting from the center of the wound and moving outwards [9,10,11]. Burn wounds, which are more fragile compared to other types, require gentler mechanical force during cleaning. Common cleaning methods include circular movements or successive, gentle pushing contacts on the wound [12]. In the case of foot ulcers, often found in individuals with diabetes, where the objective is to remove necrotic tissue, the movement should be from the innermost point outwards in a circular motion. This technique helps reduce the risk of contaminating the wound with bacteria from the surrounding skin [13,14].

The developed robotic system is performed using a Universal Robot with 6 degrees of freedom and it is capable of performing four different wound-cleaning movements, for the different types of wounds, taking into account the applied force. These movements are:

- Linear: Linear movements over the wound. (Fig. 3)
- Circular: Circular movement around the wound. (Fig. 4)
- Circular-Push: Applying pressure at different points on the wound following a circular path. (Fig. 5)
- Circular-Exterior: Outward movements along the wound's border. (Fig. 6)

Examples of these movements can be found in section 7. Graphical Explanation of the Movements

2. Tool definition

Coordinates are given with respect to the Flange frame.

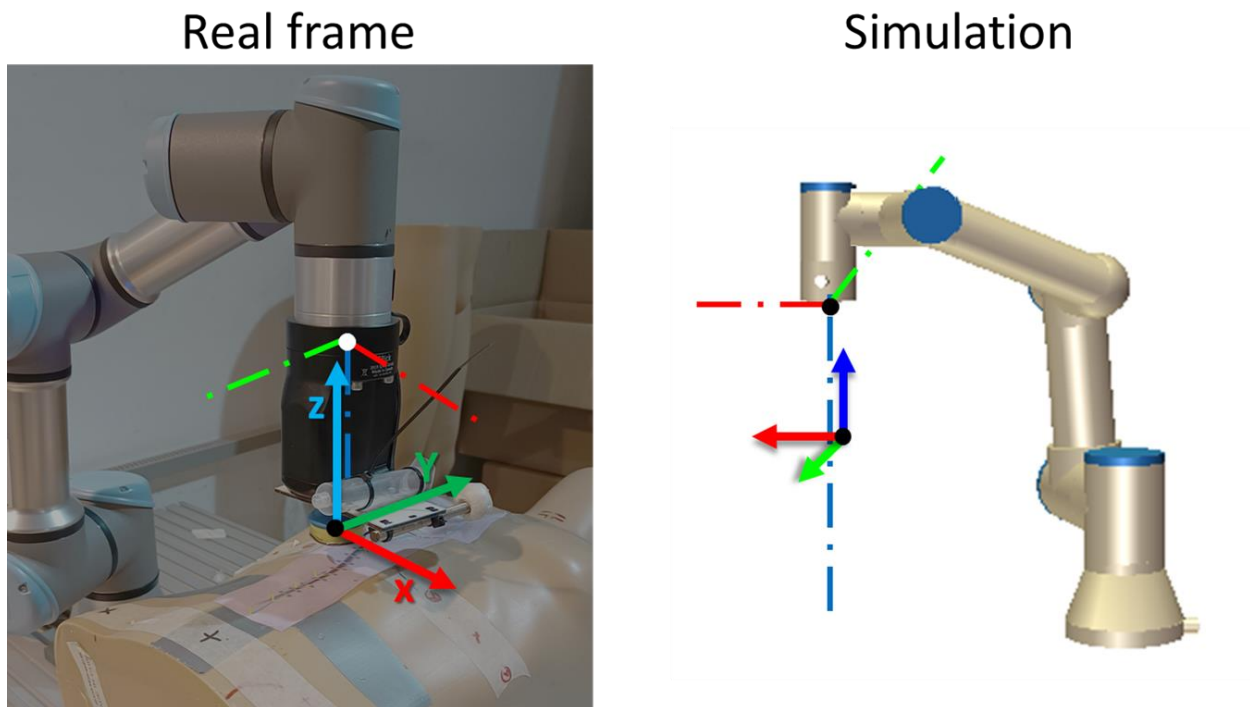| Tool name	X	Y	Z	RX	RY	RZ
Cleaner	-2.2	1.16	156.7	3.1416	0.0	0.0

Figure 1: Definition of the Tool Center Point (TCP) frame. On the left side, the frame is overlaid on the actual robot, and on the right side, the frame and robot are simulated using Universal Robots software. The reference frame (Flange frame) is illustrated with dashed lines in both images.

3. Frame definition

Coordinates are given with respect to the Base frame.

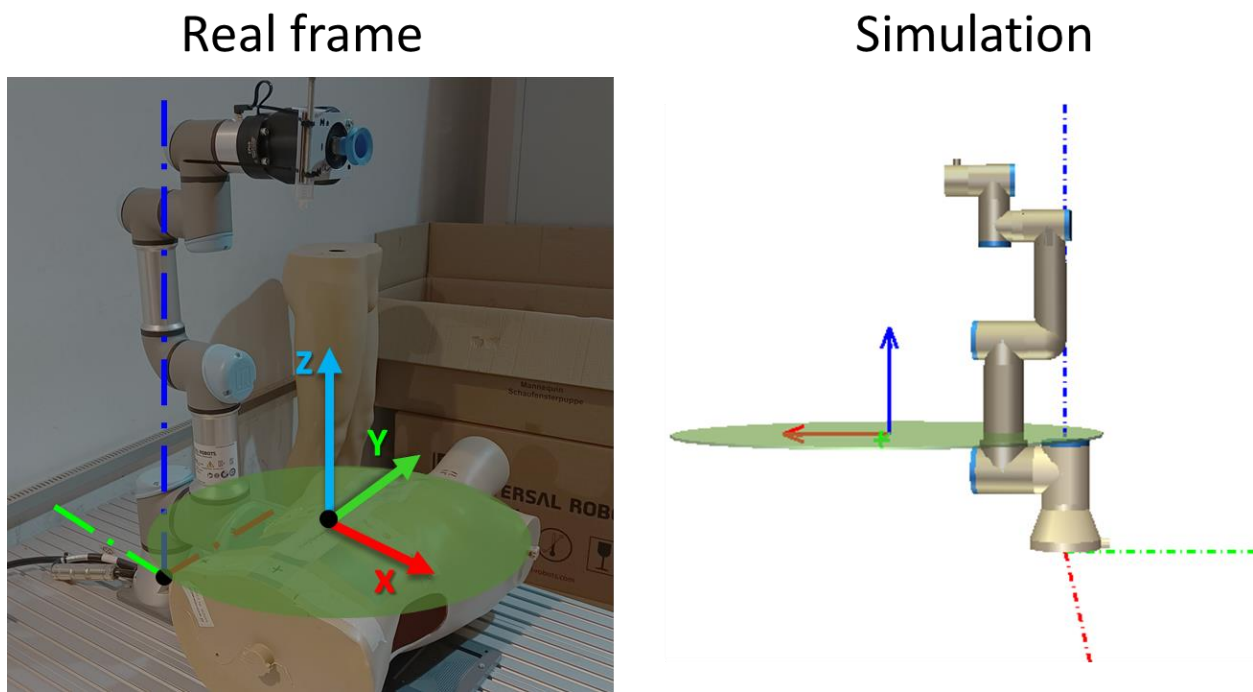| Frame name	X	Y	Z	RX	RY	RZ
origin_abd	-7.93	-328.90	220.46	0.0	0.0	4.713

Figure 2: Definition of the origin_abd frame. On the left side, the frame is overlaid on the actual robot, and on the right side, the frame and robot are simulated using Universal Robots software. The reference frame (Base frame) is illustrated with dashed lines in both images.

4. List of Variables and points

1. All coordinates mentioned are for the Tool Center Point (TCP) of the robot. These coordinates are given with respect to the frame 'origin_abd' (parent frame). 2. An additional column 'C' (Contact) is included in some tables. This column indicates whether the robot is in contact with the wound (Y) or not (N).

Global Variables

Variable	Type	Description
home_pos	Point	Home position of the robot (51,5,471)
digital_in[0]	Bool	Button 1 state (Pressed = 1)
digital_in[1]	Bool	Button 2 state (Pressed = 1)
digital_in[2]	Bool	Button 3 state (Pressed = 1)
digital_in[3]	Bool	Button 4 state (Pressed = 1)
digital_out[1]	Bool	Led 1 state (HIGH = 1)
digital_out [2]	Bool	Led 2 state (HIGH = 1)
digital_out [3]	Bool	Led 3 state (HIGH = 1)
digital_out [4]	Bool	Led 4 state (HIGH = 1)
vacuum	Bool	Vacuum stater (Suction = 1)

Take tool Movement variables

Variable	Type	Description
tool_position	Point	Position where the robot pick up the tool (-516,-317,-80)
tool_up	Point	Z up 2cm from tool_position (-516,-317,-60)

Release tool Movement variables

Variable	Type	Description
tool_release	Point	Position where the robot release the tool (-400,-317,-60)

Lineal Cleaning Movement variables

Variable	Type	Description	C
l_initial_pos	Point	Initial position of the linear movement (0,0,20)	N
l1	Point	Movement in z down from l_initial_pos (0,0,0). *initial point of the first and second line*	Y
l2	Point	Movement in y up from l1 (0,80,0). *end point of the first line*	Y
l3	Point	Movement in z up from l2 (0,80,20)	N
l4	Point	Movement in y down from l1 (0,-80,0) *end point of the second line*	Y
l3	Point	Movement in z up from l4 (0,-80,20)	N

Circular Cleaning Movement variables

Variable	Type	Description	C
c_initial_pos	Point	Initial position of the circular movement (0,0,20)	N
c1	Point	Movement in y up from c_initial_pos (0,30,20)	N
c2	Point	Movement in z down from c1 (0,30,0). *initial point of the first semi-circle and end point of the second semi-circle*	Y
c3	Point	*Second point of the first semi-circle (30, 0, 0)*	Y
c4	Point	End point of the first semi-circle and initial point of the second semi-circle (0,-30,0)	Y
c5	Point	Second point of the second semi-circle (-30,0,0)	Y

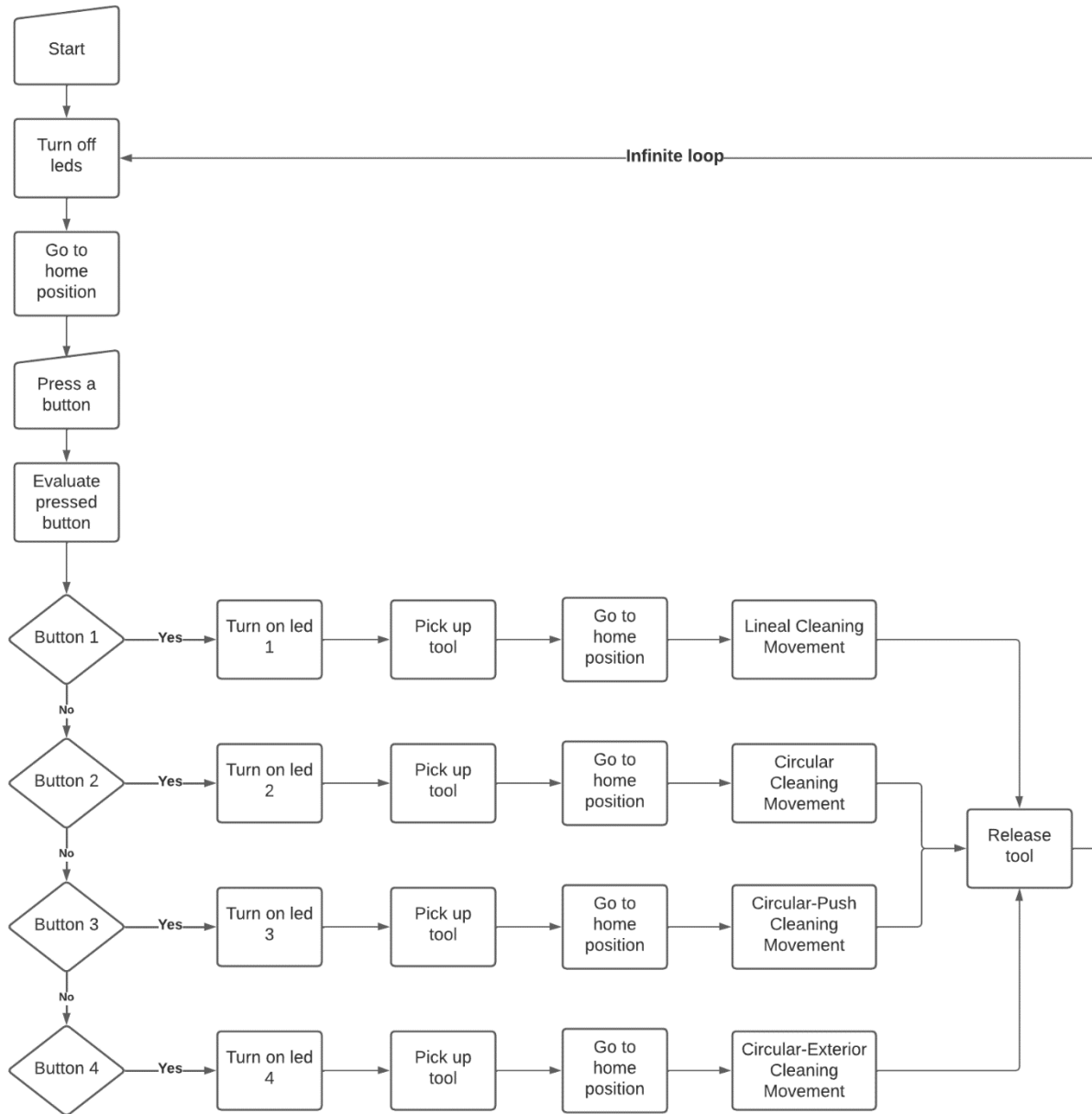## Circular-Push Cleaning Movement variables

Variable	Type	Description	C
cp_initial_pos	Point	Initial position of the circular push movement (0,0,20)	N
cp1	Point	Movement in y up from cp_initial_pos (0,30,20)	N
cp2	Point	Movement in z down from cp1 (0,30,0) *first push*	Y
cp3	Point	Movement in x up from cp_initial_pos (30,0,20)	N
cp4	Point	Movement in z down from cp3 (30,0,0) *second push*	Y
cp5	Point	Movement in y down from cp_initial_pos (0,-30,20)	N
cp6	Point	Movement in z down from cp5 (0,-30,0) *third push*	Y

| cp7 | Point | Movement in x down from cp_initial_pos (-30,0,20) | N |
| cp8 | Point | Movement in z down from cp7 (-30,0,0)
 fourth push | Y |

Circular-Exterior Cleaning Movement variables

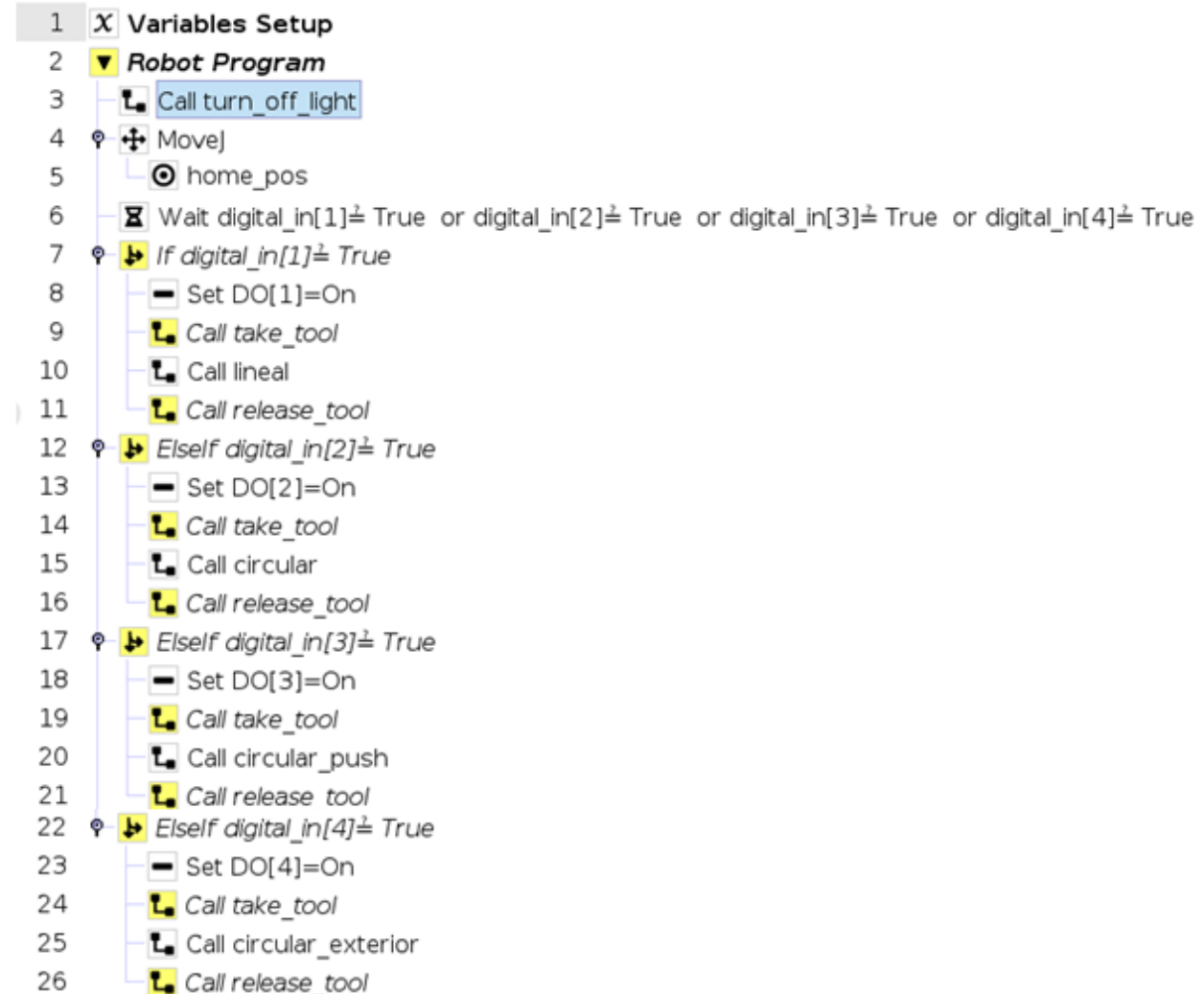| Variable	Type	Description	C
ce_initial_pos	Point	Initial position of the circular exterior movement (0,0,20)	N
ce1	Point	Movement in y up from ce_initial_pos (0,20,20)	N
ce2	Point	Movement in z down from ce1 (0,20,0) *initial point of the first exterior line*	Y
ce3	Point	Movement in y up from ce2 (0,40,0) *end point of the first exterior line*	Y
ce4	Point	Movement in z up from ce3 (0,40,20)	N
ce5	Point	Movement in x up from ce_initial_pos (20,0,20)	N
ce6	Point	Movement in z down from ce5 (20,0 ,0) *initial point of the second exterior line*	Y
ce7	Point	Movement in x up from ce6 (40,0,0) *end point of the second exterior line*	Y
ce8	Point	Movement in z up from ce7 (40,0,20)	N
ce9	Point	Movement in y down from ce_initial_pos (0,-20,20)	N
ce10	Point	Movement in z down from ce9 (0,-20,0) *initial point of the third exterior line*	Y
ce11	Point	Movement in y down from ce10 (0,-40,0) *end point of the third exterior line*	Y
ce12	Point	Movement in z up from ce11 (0,-40,20)	N
ce13	Point	Movement in x down from ce_initial_pos (-20,0,20)	N
ce14	Point	Movement in z down from ce13 (-20,0 ,0) *initial point of the fourth exterior line*	Y
ce15	Point	Movement in x down from ce14 (-40,0,0) *end point of the fourth exterior line*	Y
ce16	Point	Movement in z up from ce15 (-40,0,20)	N

5. Bloc Diagram

6. Code

Main loop and Menu

```
1   X  Variables Setup
2   ▼  Robot Program
3      Ŀ  Call turn_off_light
4   ♀ ✛  MoveJ
5         ⊙  home_pos
6      ⌛  Wait digital_in[1] ≐ True  or digital_in[2] ≐ True  or digital_in[3] ≐ True  or digital_in[4] ≐ True
7   ♀ ↳  If digital_in[1] ≐ True
8      ▬  Set DO[1]=On
9      Ŀ  Call take_tool
10     Ŀ  Call lineal
11     Ŀ  Call release_tool
12  ♀ ↳  ElseIf digital_in[2] ≐ True
13     ▬  Set DO[2]=On
14     Ŀ  Call take_tool
15     Ŀ  Call circular
16     Ŀ  Call release_tool
17  ♀ ↳  ElseIf digital_in[3] ≐ True
18     ▬  Set DO[3]=On
19     Ŀ  Call take_tool
20     Ŀ  Call circular_push
21     Ŀ  Call release_tool
22  ♀ ↳  ElseIf digital_in[4] ≐ True
23     ▬  Set DO[4]=On
24     Ŀ  Call take_tool
25     Ŀ  Call circular_exterior
26     Ŀ  Call release_tool
```

Comments:

```
Lines 1-26: Infinite loop of the program, which waits for the user to press a button to
start one of the 4 available movements.
Line 2: Initialization of the robot program.
Line 3: Calls the function to turn off all the LEDs.
Lines 4-5: Move the robot to the home position.
Line 6: Waits for the user to press a button.
Lines 7-11: Actions to perform if the user presses button 1.
      Line 8: Turns on LED 1.
      Line 9: Executes the program to take the tool.
      Line 10: Executes the program for the Lineal Cleaning Movement.
      Line 11: Executes the program to release the tool.
Lines 12-16: Actions to perform if the user presses button 2.
```
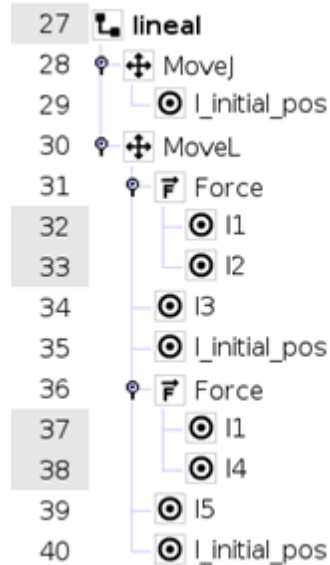
        Line 13: Turns on LED 2.
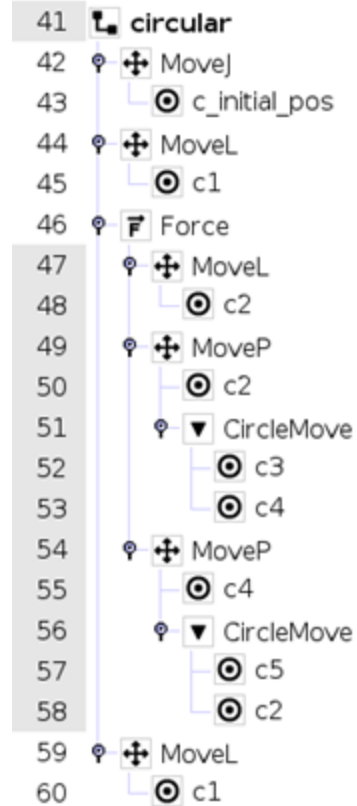   Line 15: Executes the program for the Circular Cleaning Movement.
Lines 17-21: Actions to perform if the user presses button 3.
        Line 18: Turns on LED 3.
        Line 20: Executes the program for the Circular-Push Cleaning Movement.
Lines 22-26: Actions to perform if the user presses button 4.
        Line 23: Turns on LED 4.
        Line 25: Executes the program for the Circular-Exterior Cleaning Movement

Lineal Cleaning Movement

```
27   ⌞ lineal
28   ⊕ MoveJ
29        ◉ l_initial_pos
30   ⊕ MoveL
31        𝐹 Force
32             ◉ l1
33             ◉ l2
34        ◉ l3
35        ◉ l_initial_pos
36        𝐹 Force
37             ◉ l1
38             ◉ l4
39        ◉ l5
40        ◉ l_initial_pos
```

Comments:

Line 27: Declaration of the program that executes the Lineal Cleaning Movement (LCM).
Lines 28-29: MoveJ to move the TCP to the initial position of the LCM.
Line 30: Sets all subsequent movements to use MoveL.
        Line 31: Sets the next two movements to apply a force in the z-axis (Fx) of -10 N
        and a Force Speed Limit (FSL) of 20 mm/s.
                Line 32: Executes a Slow MoveL (SML, as defined in the Movements Speed and
                Acceleration section at the end of the report) to initiate contact with the
                center of the wound.
                Line 33: Performs the first linear path of the movement using SML.
        Line 34: Utilizes a Medium MoveL (MML) to disengage from contact with the wound.
        Line 35: MML to return to the initial position.
        Line 36: Sets the next two movements to have Fx=-10 N and FSL=20 mm/s.
                Line 37: Slow MoveL to re-establish contact with the center of the wound.
                Line 38: SML to execute the second linear path of the movement.
        Line 39: MML to disengage from contact with the wound.
        Line 40: MML to return to the initial position.

Circular Cleaning Movement

```
41  ⌲ circular
42  ⊕ ✛ MoveJ
43        ⊙ c_initial_pos
44  ⊕ ✛ MoveL
45        ⊙ c1
46  ⊕ F̄ Force
47     ⊕ ✛ MoveL
48           ⊙ c2
49     ⊕ ✛ MoveP
50           ⊙ c2
51        ⊕ ▼ CircleMove
52              ⊙ c3
53              ⊙ c4
54     ⊕ ✛ MoveP
55           ⊙ c4
56        ⊕ ▼ CircleMove
57              ⊙ c5
58              ⊙ c2
59  ⊕ ✛ MoveL
60        ⊙ c1
```

Comments:

Line 41: Declaration of the program that executes the Circular Cleaning Movement (CCM).
Lines 42–43: MoveJ to move the TCP to the initial position of the CCM.
Lines 44–45: MML to move the TCP to the upper border (y-up) of the circle.
Line 46: Sets the next movements to have Fx=-10 N and FSL=30 mm/s.
      Lines 47–48: SML to initiate contact with the border of the wound.
      Lines 49–53: Slow Circular Movement (SCM) to perform the first semi-circle of the path.
      Lines 54–58: SCM to perform the second semi-circle of the path.
Lines 59–60: MML to disengage from contact with the wound.

Circular-Push Cleaning Movement

```
61   ⌐₊ circular_push
62   ⊕ ⊕ MoveJ
63      └ ⊙ ce_initial_pos
64   ⊕ ⊕ MoveL
65      ├ ⊙ cp1
66   ⊕ F̄ Force
67         ├ ⊙ cp2
68         └ ⌛ Wait: 3.0
69      ├ ⊙ cp1
70      ├ ⊙ cp3
71   ⊕ F̄ Force
72         ├ ⊙ cp4
73         └ ⌛ Wait: 3.0
74      ├ ⊙ cp3
75      ├ ⊙ cp5
76   ⊕ F̄ Force
77         ├ ⊙ cp6
78         └ ⌛ Wait: 3.0
79      ├ ⊙ cp5
80      ├ ⊙ cp7
81   ⊕ F̄ Force
82         ├ ⊙ cp8
83         └ ⌛ Wait: 3.0
84      ├ ⊙ cp7
85      └ ⊙ ce_initial_pos
```
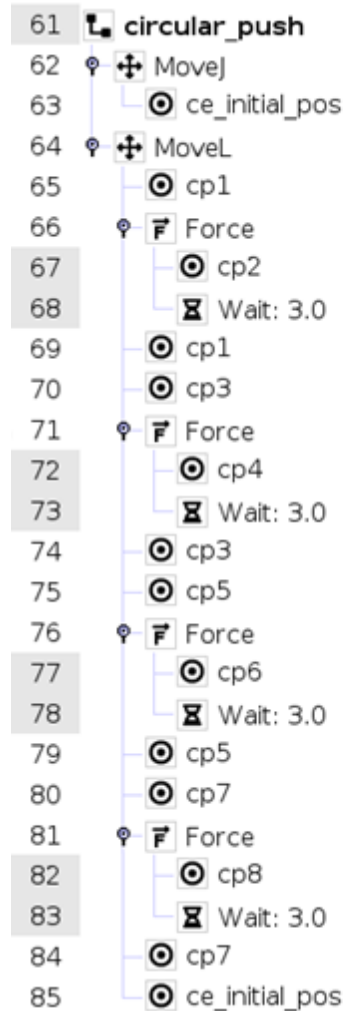
Comments:

Line 61: Declaration of the program that executes the Circular-Push Cleaning Movement (CPCM).
Lines 62-63: MoveJ to move the TCP to the initial position of the CPCM.
Line 64: Sets all subsequent movements to use MoveL.
Lines 65-69: perform the first push on the upper border of the circle (y-up).
    Line 65: MML to move the TCP to y-up.
    Line 66: Sets the next movements to have Fx=-10 N and FSL=10 mm/s.
        Line 67: SML to initiate contact with the border of the wound.
        Line 68: Wait for 3 seconds.
    Line 69: MML to disengage from contact with the wound.
Lines 70-74: perform the second push on the rightmost border of the circle (x-up).
Lines 75-79: perform the third push on the lower border of the circle (y-down).
Lines 80-84: perform the fourth push on the leftmost border of the circle (x-down).
Line 85: MML to return to the initial position.

Circular-Exterior Cleaning Movement

```
 86   ⌐ circular_exterior
 87   ⊕ ✛ MoveJ
 88        ⊙ ce_initial_pos
 89   ⊕ ✛ MoveL
 90        ⊙ ce1
 91   ⊕ F̄ Force
 92           ⊙ ce2
 93           ⊙ ce3
 94        ⊙ ce4
 95        ⊙ ce5
 96   ⊕ F̄ Force
 97           ⊙ ce6
 98           ⊙ ce7
 99        ⊙ ce8
100        ⊙ ce9
101   ⊕ F̄ Force
102           ⊙ ce10
103           ⊙ ce11
104        ⊙ ce12
105        ⊙ ce13
106   ⊕ F̄ Force
107           ⊙ ce14
108           ⊙ ce15
109        ⊙ ce16
110        ⊙ ce_initial_pos
```

Comments:

```
Line 86: Declaration of the program that executes the Circular-Exterior Cleaning
Movement (CECM).
Lines 87-88: MoveJ to move the TCP to the initial position of the CECM.
Line 89: Sets all subsequent movements to use MoveL.
Lines 90-94: perform the first circular exterior movement (CEM) on the upper part of the
circle (y-up)
      Line 90: MML to move the TCP to y-up.
      Line 91: Sets the next movements to have Fx=-10 N and FSL=30 mm/s.
             Line 92: SML to initiate contact with the internal part of the wound.
             Line 93: SML to move to the external part of the wound.
      Line 94: MML to disengage from contact with the wound.
Lines 95-99: perform the second CEM on the rightmost border of the circle (x-up).
Lines 100-104: perform the third CEM on the lower border of the circle (y-down).
Lines 105-109: perform the fourth CEM on the leftmost border of the circle (x-down).
Line 110: MML to return to the initial position.
```

Take Tool / Release Tool / Turn off lights

```
111  ⌊ take_tool
112  ⊕ MoveJ
113        ⊙ tool_up
114  F Force
115     ⊕ MoveL
116           ⊙ tool_position
117           ▬ Missing: Vacuum
118  ⊕ MoveJ
119        ⊙ home_pos
120  ⌊ release_tool
121  ⊕ MoveJ
122        ⊙ home_pos
123  ⊕ MoveJ
124        ⊙ tool_release
125     ▬ Missing: Vacuum
126  ⌊ turn_off_light
127     ▬ Set DO[1]=Off
128     ▬ Set DO[2]=Off
129     ▬ Set DO[3]=Off
130     ▬ Set DO[4]=Off
```

Comments:

```
Line 111: Declaration of the program that executes the tool-taking.
Lines 112-113: MoveJ to move the TCP above the predefined position of the tool.
Line 114: Sets the next movements to have Fx=-10 N
      Lines 115-116: FML to move in z-down towards the tool.
      Line 117: Activate vacuum (set to high) until object detected.
Lines 118-119: MoveJ to move to home position.

Line 120: Declaration of the program that executes the tool release.
Lines 121-122: MoveJ to move to home position.
Lines 132-124: MoveJ to move the TCP above the predefined position for the tool release.
Line 125: Deactivate the vacuum (set to LOW) to release the tool.

Line 126: Declaration of the program that turns off all the LEDs.
Lines 127-130: set output 1, output 2, output 3, and output 4 to LOW.
```

Movements Speed and Acceleration

<u>MoveJ:</u> All MoveJ commands have the same joint speed (JS) and joint acceleration (JA).
        JS = 60 °/s
        JA = 80 °/s²

<u>MoveL:</u>

> <u>Slow MoveL (SML) / Slow Circular Movement (SCM):</u> These movements are performed when the TCP is in contact with the patient. For safety reasons, they have a low tool speed (TS) and tool acceleration (TA).
>   TS = 10 mm/s
>   TA = 50 mm/s²
>
> <u>Medium MoveL (MML):</u> These movements are executed when the TCP is close to the patient but not in contact, allowing for faster speeds than SML while still being slow enough to prevent any harm to the patient.
>   TS = 30 mm/s
>   TA = 150 mm/s²
>
> <u>Fast MoveL (FML):</u> These movements are performed when the TCP is far from the patient, allowing for high speeds.
>   TS = 100 mm/s
>   TA = 500 mm/s²

7. Graphical explanation of the movements

Lineal Cleaning Movement
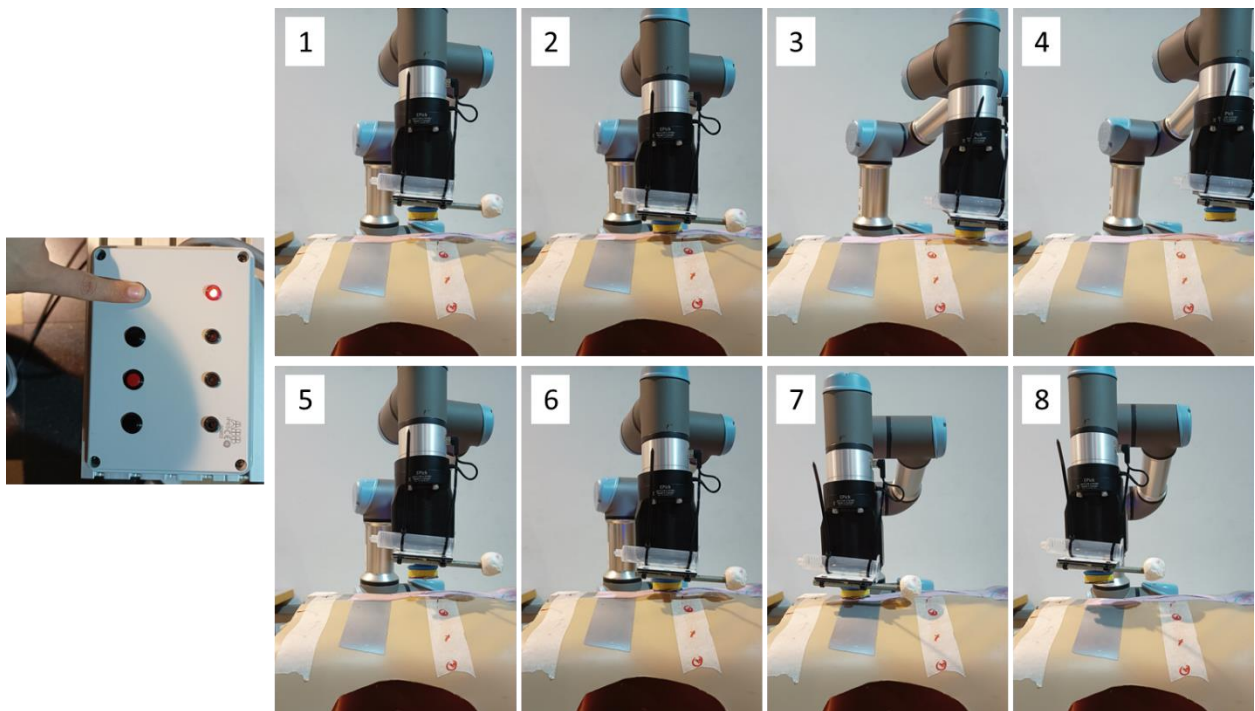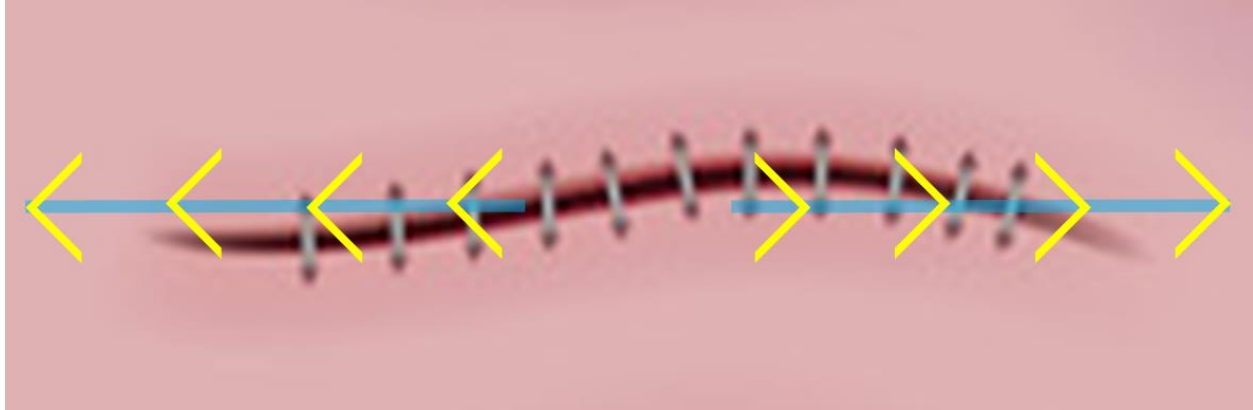
Figure 3: Graphical representation of the Linear Cleaning Movement and the steps performed by the robot.

Circular Cleaning Movement
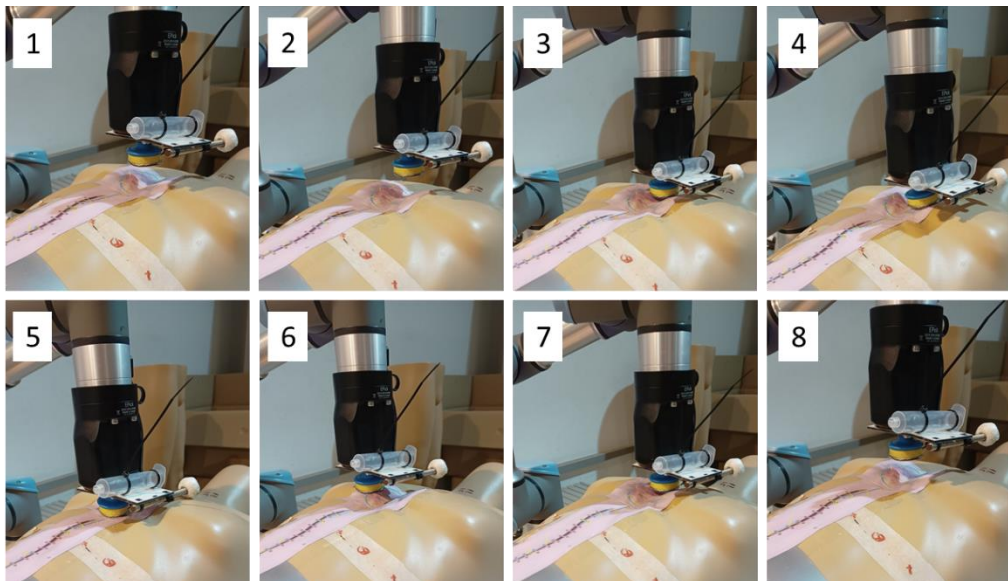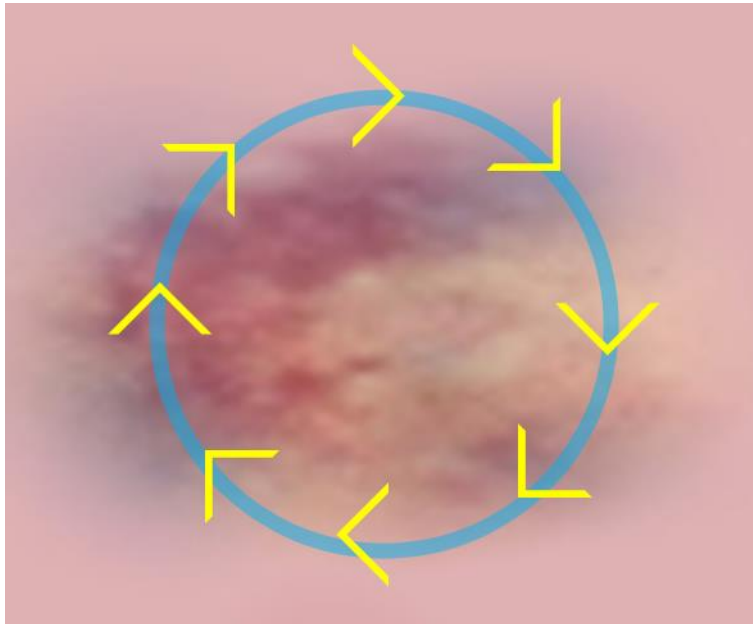
Figure 4: Graphical representation of the Circular Cleaning Movement and the steps performed by the robot.

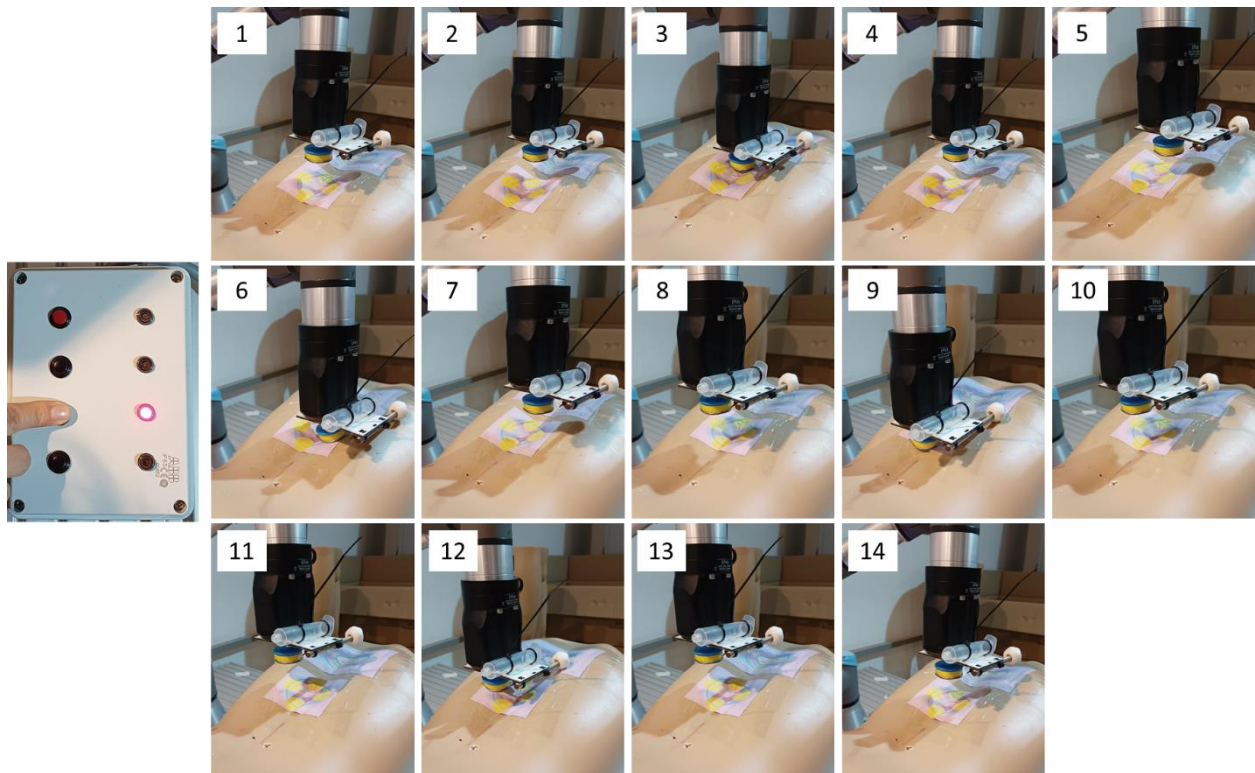## Circular-Push Cleaning Movement
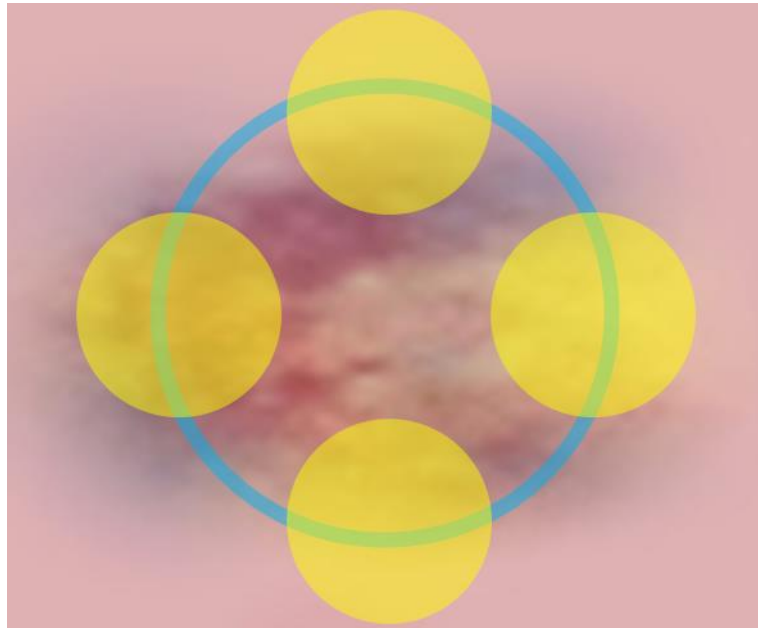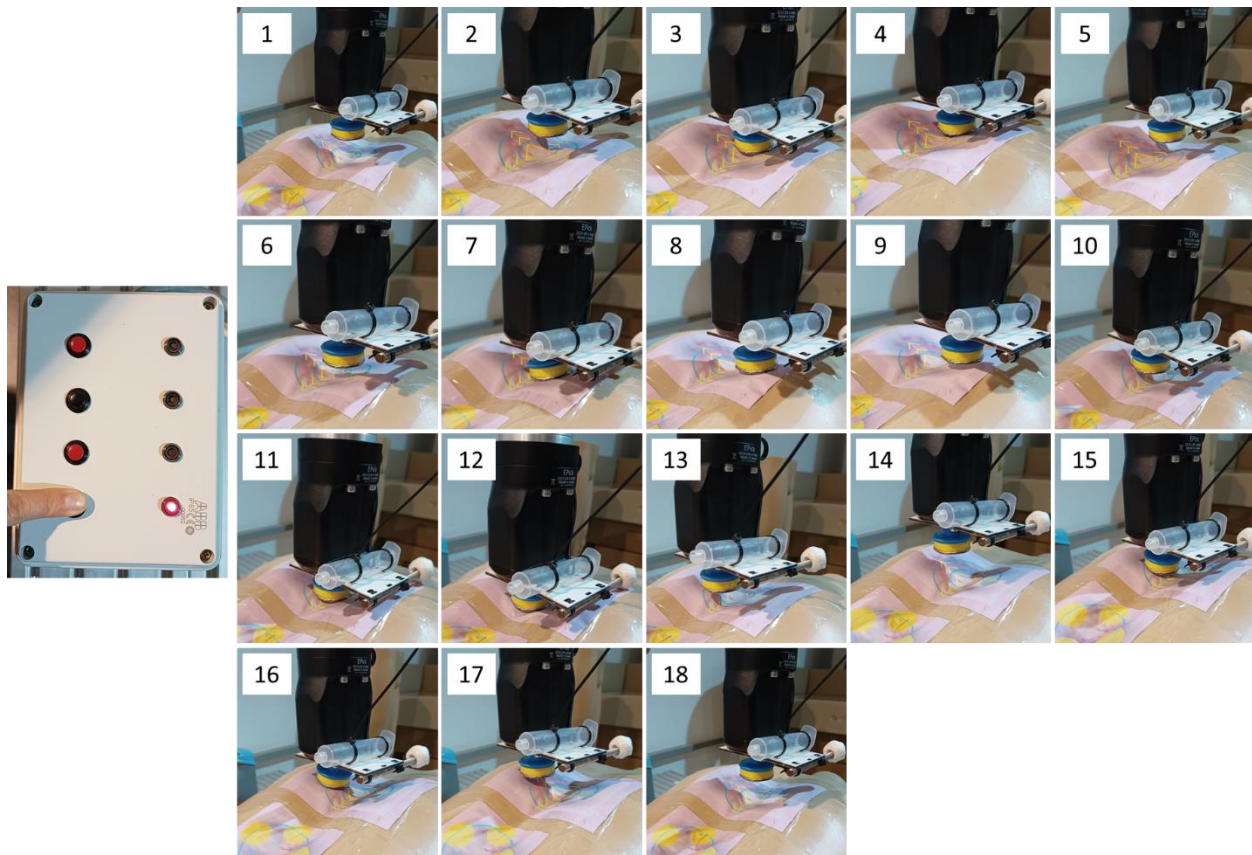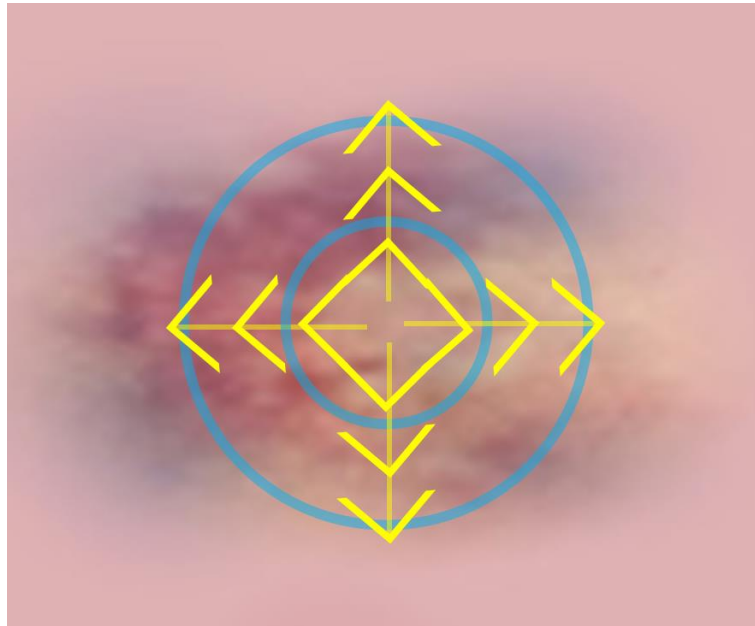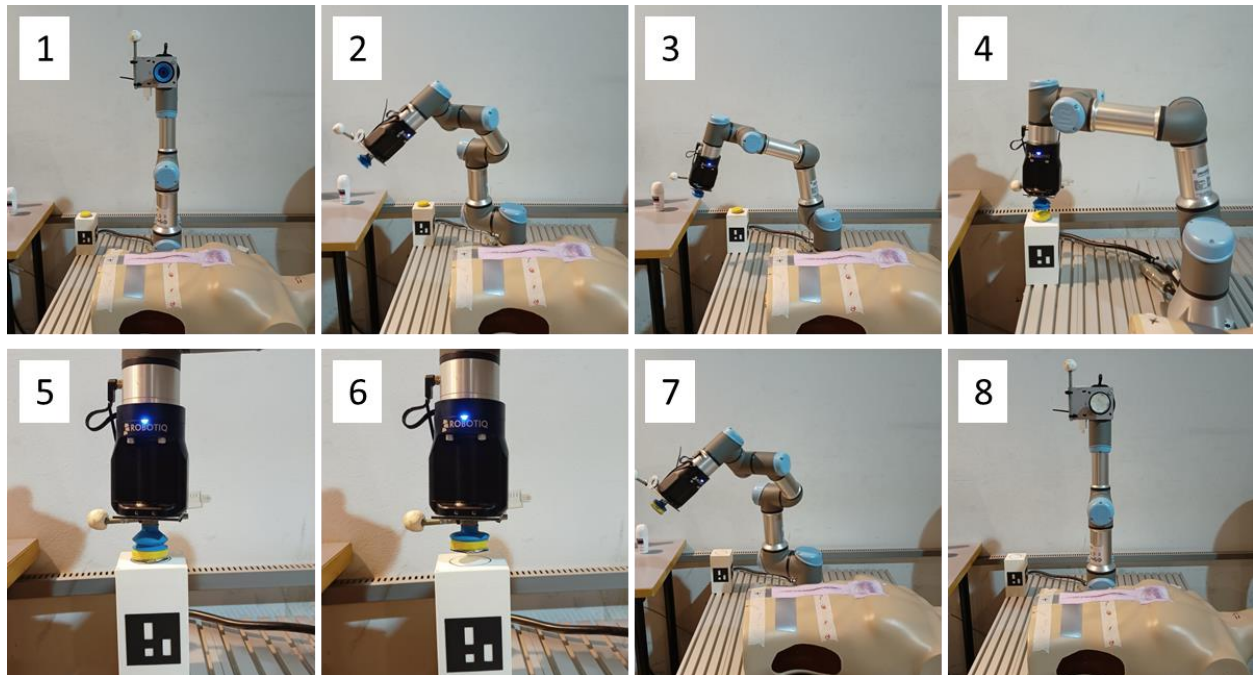

Figure 5: Graphical representation of the Circular-Push Cleaning Movement and the steps performed by the robot.

Circular-Exterior Cleaning Movement

Figure 6: Graphical representation of the Circular-Exterior Cleaning Movement and the steps performed by the robot.

Tool-Taking

Figure 7: Steps performed by the robot to execute the tool-taking process.
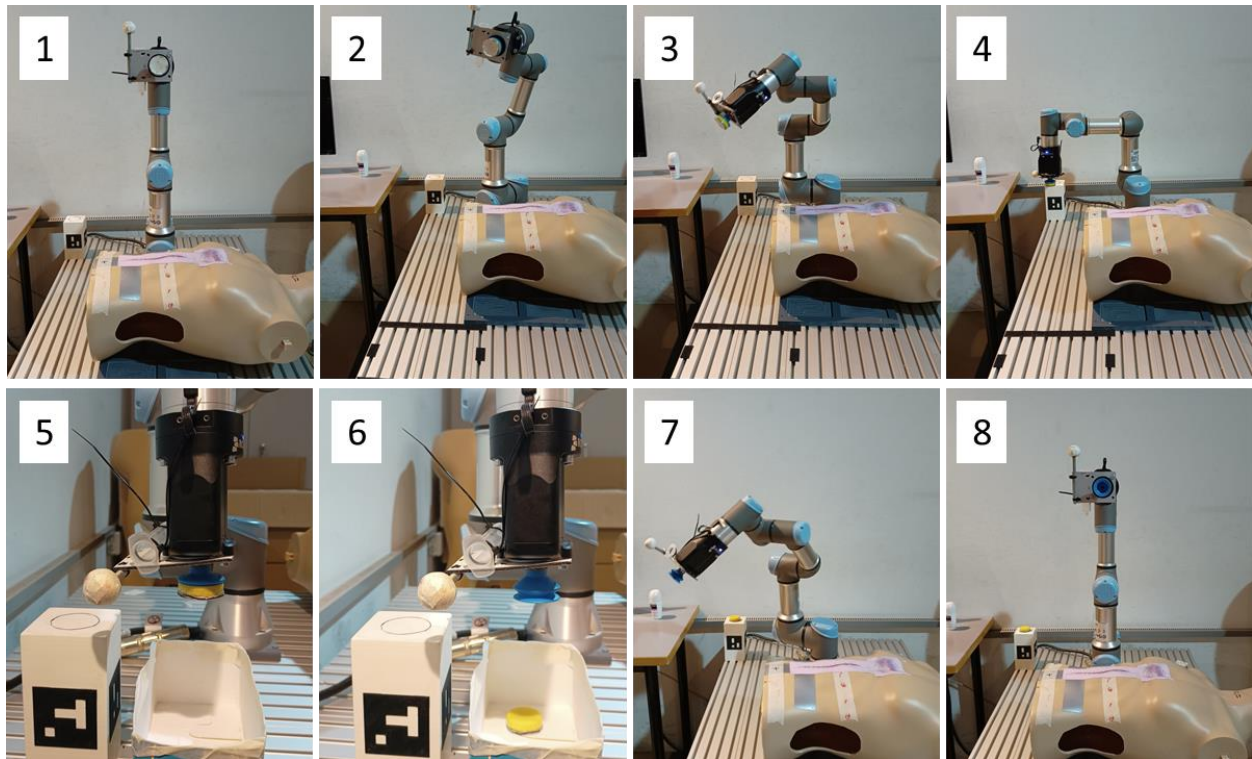
Tool-Release

Figure 8: Steps performed by the robot to execute the tool-release process.

8. Bibliography

1. Cepolina F, Razzoli RP. Review of Robotically Assisted Surgical Systems. Int J Med Robot. 2022;18(4):e2409. doi: 10.1002/rcs.2409.
2. Klodmann J, et al. Introduction to Robotically Assisted Surgical Systems. Curr Robot Rep. 2021;2:321-332. https://doi.org/10.1007/s43154-021-00064-3
3. Boubaker O. Medical Robotics. In: Academic Press; 2020. pp. 153-204. https://www.sciencedirect.com/science/article/pii/B978012821350600007X
4. Rivero-Moreno Y, et al. Robotic Surgery: Comprehensive Literature Review. Cureus. 2023;15(7):e42370. doi: 10.7759/cureus.42370.
5. https://www.intel.com/content/www/us/en/healthcare-it/robotics-in-healthcare.html
6. https://www.kindredhospitals.com/our-services/ltac/conditions/wound-care#:~:text=Wounds%20can%20be%20open%2C%20with,Puncture%20wounds
7. McLain NEM, Moore ZEH, Avsar P. Wound cleansing for treating venous leg ulcers. Cochrane Database of Systematic Reviews 2021, Issue 3. Art. No.: CD011675. DOI: 10.1002/14651858
8. https://www.mountsinai.org/health-library/selfcare-instructions/surgical-wound-care-open
9. https://medlineplus.gov/ency/patientinstructions/000040.htm
10. https://www.youtube.com/watch?v=c-S7SHeHfrs
11. https://blog.wcei.net/top-tips-for-wound-cleansing
12. https://www.physio-pedia.com/Burn_Wound_Treatment:_Cleansing_and_Solutions
13. https://www.naxlex.com/nursing/question/which-technique-is-most-appropriate-for-a-nurse-to-use-when-cleansing
14. https://www.physio-pedia.com/Wound_Debridement