


Baxter_2022

Contents

Baxter, The Bionic Robot The Edge	3:23	316,000 views	Rodney	1
Baxter the Robot Making Coffee - Alexa Interface				1
Baxter Controlled through EEG	3:13			2
Using a state machine to perform YMCA				2
PYTHON Baxter YMCA		Be Careful because of spacing – see the code.		3
See Baxter YMCA Code				4
BAXTER PLAYS GOLF				5

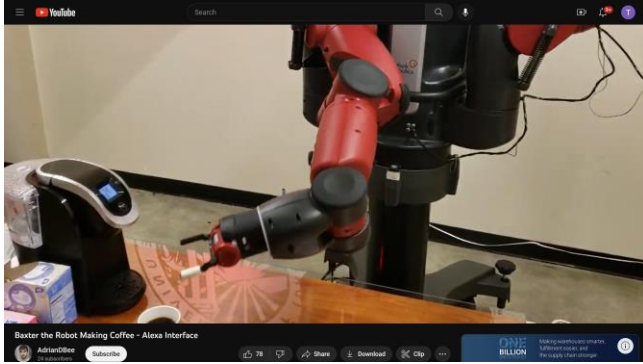
Baxter, The Bionic Robot | The Edge **3:23** **316,000 views** **Rodney**

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



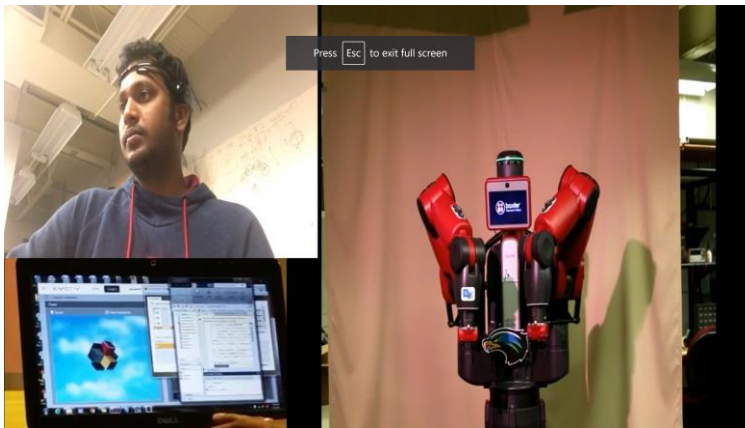
Baxter the Robot Making Coffee - Alexa Interface

<https://www.youtube.com/watch?v=kO5B5dSiI9A>



Baxter Controlled through EEG 3:13

<https://www.youtube.com/watch?v=RSYHQQoSIR8>



See Our Textbook Pages 303 -308 State Machine YMCA

Using a state machine to perform YMCA

Finite-state machines are powerful mechanisms for controlling the behavior of a system, especially robotic systems. ROS has implemented a state machine structure and behaviors in a Python-based library called **SMACH**. The SMACH library is independent of ROS and can be used with any Python project. SMACH provides an architecture for implementing hierarchical tasks and mechanisms to define transitions between these tasks. The advantages of using SMACH for a system include the following:

- Rapid prototyping of a state machine for testing and use
- Defining complex behaviors using a clear, straightforward method for design, maintenance, and debugging
- Introspection of the state machine, its transitions, and data flow using SMACH tools

For a complete set of documentation and tutorials on SMACH, examine these websites:


- <http://wiki.ros.org/smach>
- <http://wiki.ros.org/smach/Tutorials>

Some basic rules for implementing state machines on a robot are as follows:

- A robot can be in one – and only one – state at a time.
- A finite number of states must be identified.
- The state that a robot transitions to, will depend on the state that just completed. These behaviors are encapsulated in the states to which they correspond.
- Transitions between states are specified by the structure of the state machine.
- All possible outcomes of a state should be identified and corresponding behaviors should address those outcomes.
- States that only have one transition condition cannot fail and only have one outcome.

To underscore the usefulness of the SMACH package, we devised a simple and fun example for Baxter, which has been implemented by Mikal Cristen, a recent UHCL graduate. Because the UHCL Baxter is such a main attraction on our campus, this project was to endow the robot with an entertainment skill, specifically, dancing to YMCA.

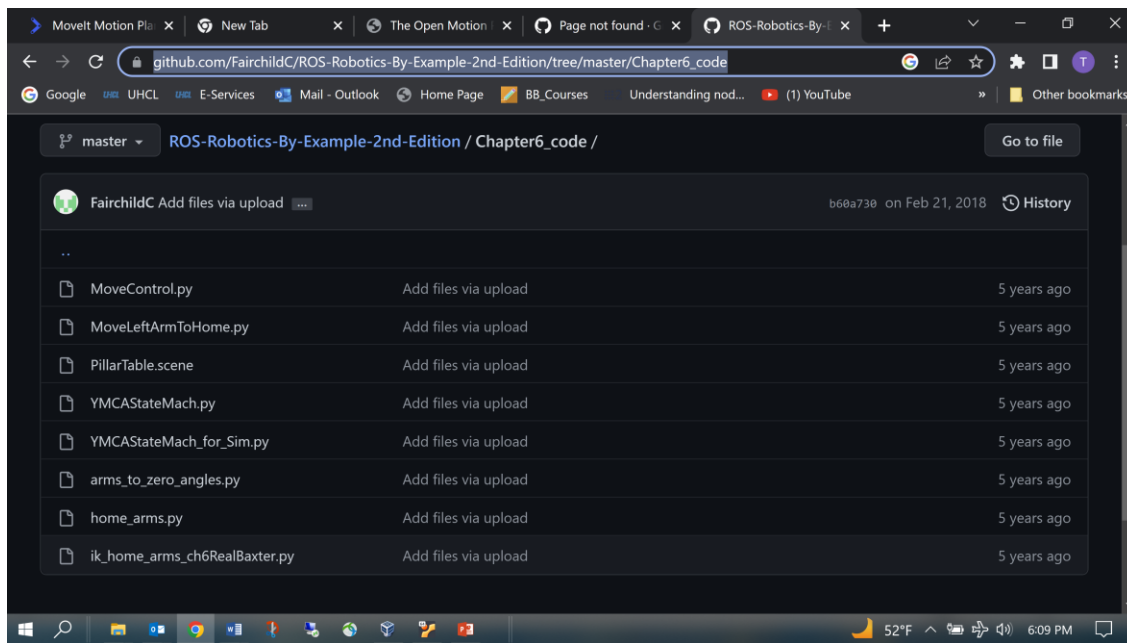
This state machine has five states corresponding to Baxter's arm poses for each letter: Y, M, C, A, and a fifth state for a neutral pose. When one pose of the arms completes, the state will successfully complete and the next state will begin. The code for this state machine is implemented in the `YMCAStateMach.py` code that follows and will be described in subsequent paragraphs.


 The code for `YMCAStateMach.py` and `MoveControl.py` can be found in the Chapter 6 folder of the Packt GitHub website for this book or at the website [https://github.com/ FairchildC/ROS-Robotics-By-Example-2nd-Edition](https://github.com/FairchildC/ROS-Robotics-By-Example-2nd-Edition)

SMACH compels state machines to be implemented using Python procedures to provide flexibility in their implementation. Notice in the following code, the ROS convention for state machines is that the state names are identified in ALL_CAPS and the transition names are in lowercase.

PYTHON Baxter YMCA  **Be Careful because of spacing – see the code.**

https://github.com/FairchildC/ROS-Robotics-By-Example-2nd-Edition/tree/master/Chapter6_code



See Baxter YMCA Code

https://github.com/FairchildC/ROS-Robotics-By-Example-2nd-Edition/tree/master/Chapter6_code

BAXTER PLAYS GOLF

https://sdk.rethinkrobotics.com/wiki/Worked_Example_Visual_Servoing#Code



Baxter Research Robot Visual Servoing Pick & Place Demonstration 3:55

<https://www.youtube.com/watch?v=d-a82DYlaIE>