Choosing a processing device

Aim

Control the actuators based on the inputs from either the driver control input device, or the sensors.

Objectives

Functions Objectives

1. Allow control during the driver control period.
2. Take control during the Autonomous period.
3. Control the actuators.
4. Take readings from the chosen sensors.

Non functional objectives

5. Comply with the rules.
6. Be complex enough to achieve marks in the VEX U category
7. Be achievable, can it be build and programmed easily.

Concepts

- A. Raspberry PI. use a Raspberry pi 4 with Raspbian and python, with its own drive motors and drive input control.
- B. Raspberry pi to feed into the V5 brain
- C. V5 brain. use the V5 brain with VEX motors and driver input device.

A. Raspberry Pi 4 - with bespoke actuators and driver input device.

Pros	Conns
	Does not comply with the rules
Easy to program in python	Not compatible with the VEX Battery
Opens the door to the use of non VEX sensors and actuators.	Not compatible with the VEX Motors
	Not compatible with the VEX driver input

Pros	Conns
	Open loop Autonomous mode is harder to achieve than using the VEX V5 allone
Development can progress outside of university weeks	

Compatibility with the Objectives

1. In its currents state, a mechanism would need to be made to allow the driver to control the robot, this is about 2 or 3 hours work.
2. Python is easy enough to program, however depending on the autonomous mechanism (open loop, closed loop) this will take a lot of time, 10+ hours.
3. It is easy enough to output to actuators, however for high load actuators such as drive motors, additional circuitry would needed, this is about 3 to 4 hours work.
4. reading from sensors is possible.
5. It does not comply with the rules <VUR12> Sensors and Additional Electronics MUST be connected to the V5 Robot Brain...
6. This is a very complex route and will comply with this objective.
7. This is not very achievable, many hours will have to be used to develop the driver input, the drive actuator circuit.

Summary

This is not compliant with the rules.
This will take too much time to develop.

B. Raspberry Pi 4 - Feeding into the V5 brain.

Pros	Conns
Complies with the rules	It is unknown how to talk to the V5 VEX
Easy to program in python	
Opens the door to the use of non VEX sensors and actuators.	
The VEX driver input mode can be used	
THE vex drive motors can be used	
Some code can be tested outside of university weeks	

Compatibility with the Objectives

1. The VEX controller can be used during driver input mode
2. Python is easy enough to program, however depending on the autonomous mechanism (open loop, closed loop) this will take a lot of time, 10+ hours.
3. The VEX Motors can be used
4. reading from sensors is possible.
5. It complies with the rules
6. This is a complex route and will comply with this objective.
7. There is still an unknown as to how the RPI talks with the V5 brain. Assuming the communication is possible it is achievable.

Summary

This concept is a good choice, however before deciding on this, a proof of concept investigation should take place as to whether it is possible to control the V5 brain with a RPI 4.

C. Use the V5 Brain and VEX sensors and actuators.

Pros	Conns
Complies with the rules	It is not very complex and a full marks might not be awarded.
Easy to use the block programmer	Limited to VEX sensors
	code cannot be tested outside of university weeks
The VEX driver input mode can be used	
THE vex drive motors can be used	

Compatibility with the Objectives

1. The VEX controller can be used during driver input mode
2. Easy to program.
3. The VEX Motors can be used
4. Limited to the VEX sensors
5. It complies with the rules
6. This is not very complex and might not award full marks
7. If we are limited to current sensors, it might not be possible to complete this project.

Summary

This is a good Plan B if concept B does not work.

Unanswered questions

- Are we limited to the VEX sensors we have, or can we purchase additional VEX sensors?

https://www.vexforum.com/t/v5-brain-to-raspberry-pi-communication/124407