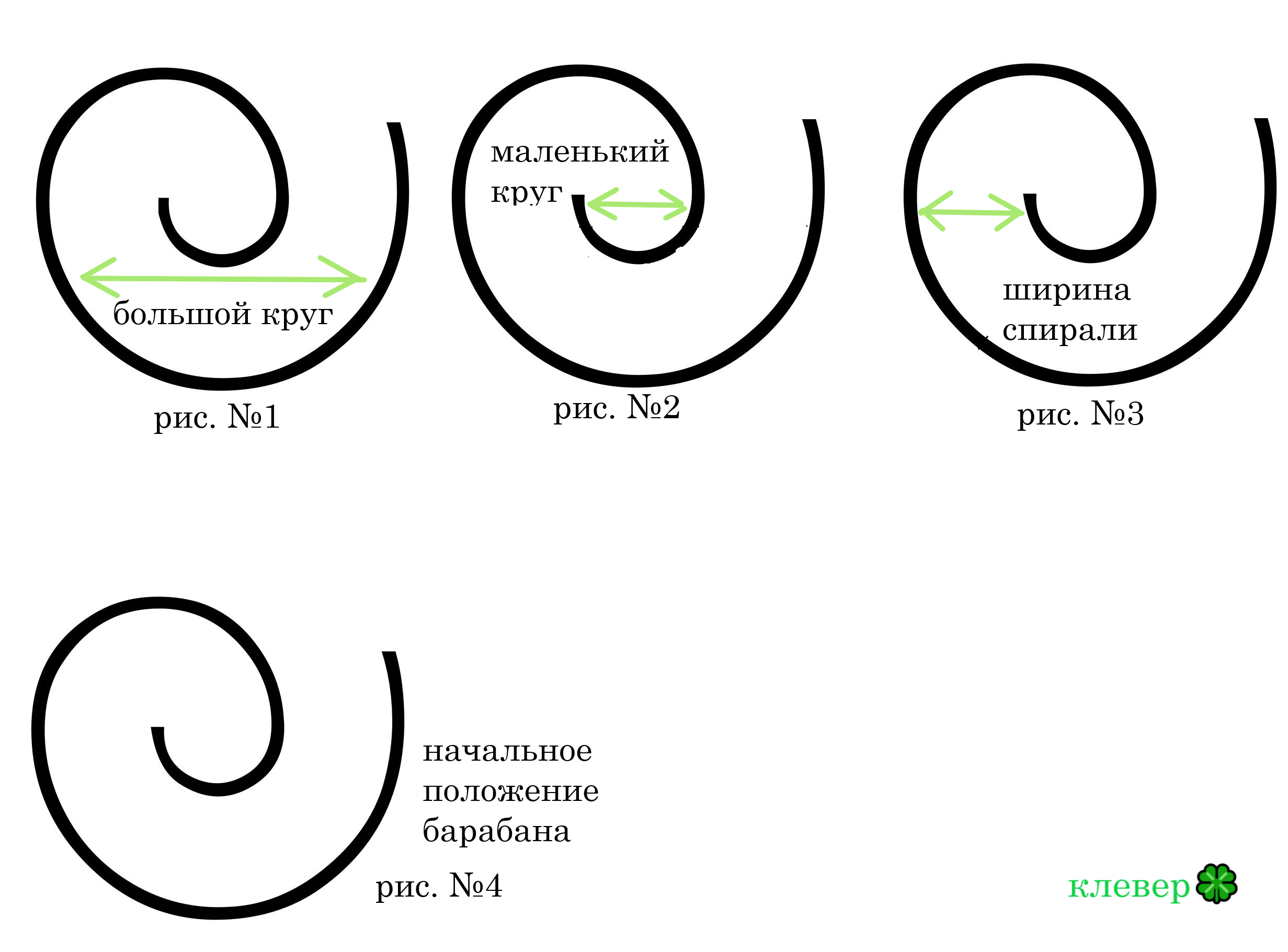
# Реализация узлов

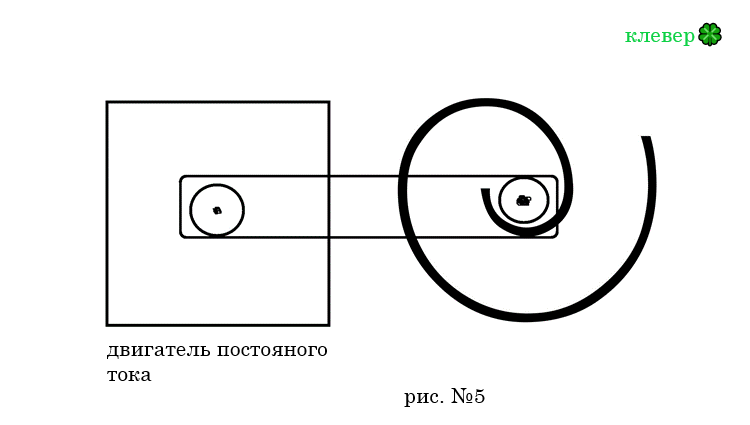
* Барабан для перемешивания игральных костей:

Сам барабан представляет собой спиралевидную конструкцию с характеристиками: диаметр большого круга 12 см (рис. №1), диаметр малого круга 8 см (рис. №2), ширина спирали 4 см (рис. №3), материал корпуса - из оргстекла (акрил), по бокам барабан закрыт прозрачными стенками из пластика, соединенными с башней свободного падения. Он расположен так, чтобы после перемешивания, проход для игральных костей располагался ровно над проходом в башню свободного падения.

Начальное и конечное положение системы (рис. №4) предполагает возможность добавление игральных костей через верхнюю часть конструкции напрямую в барабан.



С помощью двигателя постоянного тока, управляемого с помощью Arduino, происходит прокрутка барабана благодаря чему кости перемешиваются, а форма спирали не позволяет игральным костям выпасть: ударяясь о внутреннюю стенку кости не вылетают. Затем прокруткой барабана тем же двигателем постоянного тока в обратную сторону происходит выгрузка кубиков в башню свободного падения: кости проскальзывая по внешней стенке спирали, дойдя до конца спирали, вываливаются наружу в башню свободного падения. Дальше барабан возвращается в начальное положение. Двигатель постоянного тока расположен за барабаном на башне свободного падения, он приводит в движение барабан с помощью ременного механизма: с одной стороны ремень крепится к двигателю, а с другой - к шестеренке, расположенной по центру сбоку от барабана.



* **Башня свободного падения:**

Лепестки расположены спиралевидной лестницей. Попав в башню, игральные кости перемещаются по ступенькам спускаясь к основанию. Башня сделана из лестницы напечатанный на 3Д принтере и прозрачной тубы.

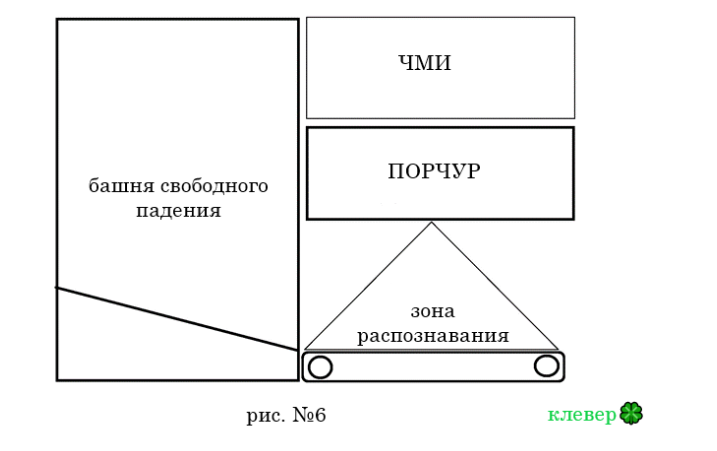
Гарантирует попадание игральных костей в пространство отведенное под подсистему оптического распознавания численного результата (ПОРЧР). После выпадения, игральные кости передаются в ПОРЧР на конвейере. С помощью следующих действий**:**

Из башни свободного падения игральные кости выпадают на конвейер, конвейер представляет собой ременной механизм, с микро мотором редуктором (GA12-N20) постоянного тока, управляемые Arduino, соединяющий башню свободного падения и ПОРЧР, при этом по бокам от конвейера расположены съемные прозрачные стенки, чтобы предотвратить возможность выпадения игральных костей. (рис.№6)

* **Подсистема оптического распознавания численного результата, выпавшего на игральных костях:**

ПОРЧР включает в себя камеру, расположенную над конвейером, подключенную к плате Raspberry Pi. Камера делает снимки игральных костей, расположенных на конвейере, затем передает их в обработку, которая происходит с помощью кода, написанного на Python библиотеке OpenCV. Далее полученные результаты отправляются в напрямую подключенный человеко-машинный интерфейс (ЧМИ) и в удаленную базу данных (БД).

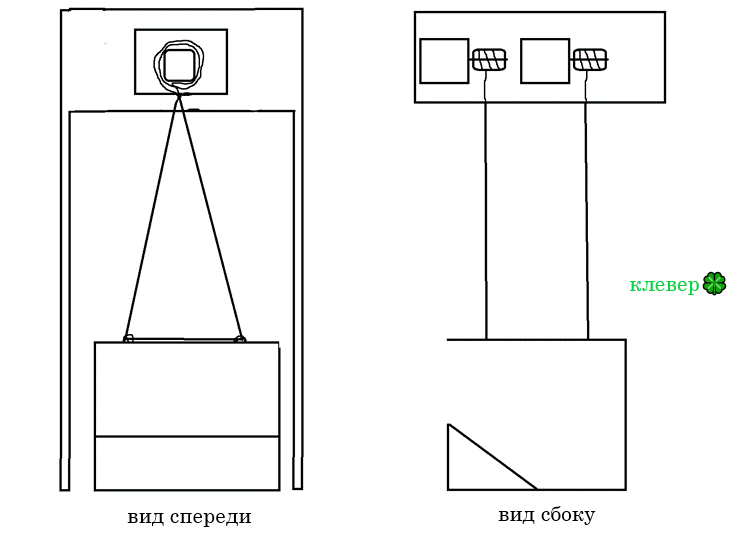
ПОРЧР находится в коробочке, которая крепится сбоку к башне свободного падения так, чтобы камера наблюдала зону, в которую попадают игральные кости после выпадения из башни свободного падения.



* **Подсистема возврата игральных костей из области распознавания в подсистему перемешивания:**

Двигаясь по тому же конвейеру, игральные кости падают в прозрачный контейнер, который закреплен на трос к шаговому двигателю (17HE15-1504S с драйвером TMC2208), управление которыми происходит через Arduino. Двигатель поднимает контейнер до уровня следующего конвейера, аналогичного первому, двигатель останавливается на достигнутом уровне. После встроенный в контейнер сервопривод поднимающей платформу на которой лежат игральные кости создавая скат для их перемещения на конвейер. Затем контейнер опускается обратно в исходное положение.

Контейнер имеет размеры в ширину 8 см, в длину 6 см, в высоту 7 см, нижняя сторона треугольника в контейнере 2 см, левая сторона треугольника 4 см.



После того, как игральные кости проедут на втором конвейере, они упадут обратно в барабан перемешивания.

* **Подсистема человеко-машинного интерфейса:**

Ч.М.И. является дисплеем ардуино. С его помощью происходит мониторинг результатов всех бросков, а также просмотр истории бросков, хранящихся в удаленной БД. Реализует возможность пользователя управлять бросками и настраивать их количество. Также с ее помощью пользователь может запускать и проверять каждую подсистему отдельно от всего программно-аппаратного комплекса. Полностью прекращать его работу осуществляя аварийную остановку. Связь со всеми подсистемами реализована через Arduino напрямую. Интерфейс расположен над ПОРЧР и управляется с помощью сенсора.