# ***Stepper Library***

***Constructor:***

- Stepper()
  - Creates a new Object called „Stepper"
  - Is made for a4988 driver
  - Use init()-method to initialize the driver

***Methods:***

- boolean **getEnabled**()
  - Get enabled-state of driver

- boolean **getOnPosition**()
  - Check if driver is on position

- byte **getDir**()
  - Get direction of driver

- double **getPosition**()
  - Get the driven distance

- boolean **halfStep**()
  - Make a half step (change step state from low to high or from high to low
  - This function must be called in a loop without or with minimal delay
  - Delay with accelration and braking is integrated
  - Returns true if steps can be made

- void **init**(byte stepPin, byte dirPin, byte enablePin)
  - Initialize driver by pins step, dir and enable.

- void **setAccelration**(float accelration)
  - Set steppers accelration in m/(s*s)

- void **setDiameter**(float diameter)
  - Set wheel diameter in mm

- void **setDir**(byte direction)
  - Set direction of driver

- void **setEnabled**(boolean enabled)
  - Set enabled state for driver

- void **setFeedrate**(float feedrate)
  - Set feedrate in m/s

- void **setPosition**(double position)
  - Set position to drive in mm

- void **setStepsPerRotation**(int stepsPerRotation)
  - Set steps per rotation
  - Don't use steps per rotation from steppers datasheet
  - You have to calculate with your driver configuration

- void **setStepState**(boolean state)
  - Set driver state to low or high
  - Can be used to control stepper manually

- void **setStopFeedrate**(float feedrate)
  - Set minimum feedrate by braking
  - After reaching this feedrate, the movement will be stopped