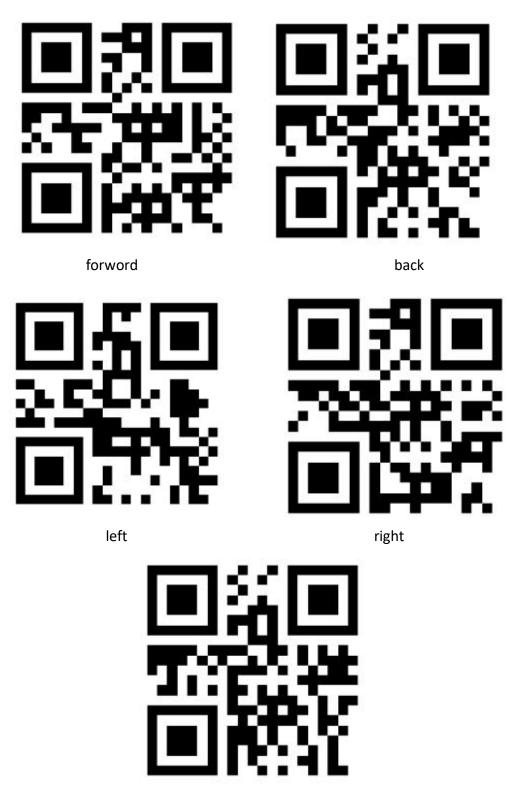### 3.1.6 QR recognition+Movement

As shown in the figure below, the five QR codes correspond to different functions and can be used to control the movement of the robot.

forword

back

left

right

Stop

/home/pi/Yahboom_Project/Raspbot/3.AI Vision course/07.QR code control/QR code control.ipynb

```python
#bgr8 to jpeg
import enum
import cv2

def bgr8_to_jpeg(value, quality=75):
    return bytes(cv2.imencode('.jpg', value)[1])
```

```python
# import the necessary packages
#import simple_barcode_detection
import cv2
import numpy as np
import pyzbar.pyzbar as pyzbar
from PIL import Image
import ipywidgets.widgets as widgets

import YB_Pcb_Car
car = YB_Pcb_Car.YB_Pcb_Car()

image_widget = widgets.Image(format='jpeg', width=320, height=240)
display(image_widget)
```

```python
def detect_control(info):
    if info == "forward":
        car.Car_Run(60,60)              # advance
    elif info == "back":
        car.Car_Back(60,60)             #back
    elif info == "left":
        car.Car_Spin_Left(60,60)        #spin left
    elif info == "right":
        car.Car_Spin_Right(60,60)       #spin right
    elif info == "stop":                #stop
        car.Car_Stop()
```

```python
def decodeDisplay(image):
    barcodes = pyzbar.decode(image)
    for barcode in barcodes:
        (x, y, w, h) = barcode.rect
        cv2.rectangle(image, (x, y), (x + w, y + h), (225, 225, 225), 2)
        barcodeData = barcode.data.decode("utf-8")
        barcodeType = barcode.type
```

```
            text = "{} ({})".format(barcodeData, barcodeType)
            cv2.putText(image, text, (x, y - 10), cv2.FONT_HERSHEY_SIMPLEX, 0.5, (225,
225, 225), 2)

            print("[INFO] Found {} barcode: {}".format(barcodeType, barcodeData))
            detect_control(barcodeData)
    return image

def detect():
    camera = cv2.VideoCapture(0)
    camera.set(3, 320)
    camera.set(4, 240)
    camera.set(5, 30)
    # fourcc = cv2.VideoWriter_fourcc(*"MPEG")
    camera.set(cv2.CAP_PROP_FOURCC, cv2.VideoWriter.fourcc('M', 'J', 'P', 'G'))
    camera.set(cv2.CAP_PROP_BRIGHTNESS, 40)
    camera.set(cv2.CAP_PROP_CONTRAST, 50)
    camera.set(cv2.CAP_PROP_EXPOSURE, 156)
    ret, frame = camera.read()
    image_widget.value = bgr8_to_jpeg(frame)
    try:
        while True:
            ret, frame = camera.read()
            gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
            im = decodeDisplay(gray)
            image_widget.value = bgr8_to_jpeg(im)
            cv2.waitKey(5)
    except:
        camera.release()
```

```
detect()
car.Car_Stop()
del car
print("Ending")
```