

```

# Spurenapparat auf Klein

from machine import Pin
led = Pin(25, Pin.OUT)
led.high(); machine.lightsleep(10); led.low()

print("Start display")
from driver2in9 import EPD_2in9, EPD_WIDTH, EPD_HEIGHT
epd = EPD_2in9()
epd.fill(0xff)
epd.display_Base(epd.buffer)

print("Sleep 5 seconds") # Pause mit weißem Grund
machine.lightsleep(5*1000) # vor langer Lagerung jetzt Stecker ziehen
led.high(); machine.lightsleep(10); led.low()

print("Prep move: WxH=%dxd" % (EPD_WIDTH, EPD_HEIGHT))
from random import randint
numCols = randint(2, 8)
numRows = randint(2, 8)
cellWidth = int(EPD_HEIGHT / numCols) # Height and
cellHeight = int(EPD_WIDTH / numRows) # width swapped !
maxStp = min(20, int(cellWidth / 3)) # Tempolimit

print("Prep move: RxC=%dx%d+%d" % (numRows, numCols, maxStp))
from array import array
i = array('i'); r = array('i'); d = array('i')
for a in range(0, numRows):
    i.append(0)
    r.append(a % 2)
    d.append(randint(1, maxStp))
x = 0

print("Start moving")
for z in range(0, 50):

    for j in range(0, numRows):
        y = j * cellHeight

        if (i[j] + d[j]) < cellWidth:
            i[j] += d[j]
        else:
            i[j] = 0
            d[j] = randint(1, maxStp)

        for k in range(0, numCols+1):
            if k == 0:
                x = 0
                v = i[j]
            elif k == 1:
                x += i[j]
                v = cellWidth
            else:
                x += cellWidth
                if k == numCols:
                    v = cellWidth - i[j]
                else:
                    v = cellWidth

            # FrameBuffer.fill_rect(x, y, w, h, c)
            if k % 2 == r[j]:
                epd.fill_rect(y, x, cellHeight, v, 0xff) # h/w swapped !
            else:
                epd.fill_rect(y, x, cellHeight, v, 0x00) # h/w swapped !

            if (i[j] + d[j]) >= cellWidth: r[j] = int(not r[j])
            epd.display_Partial(epd.buffer)
        print("%3d" % z)

print("Sleep")
epd.sleep() # Schirm stromlos schalten
machine.deepsleep(5*60*1000) # (5) Minuten warten
machine.soft_reset() # Schirm vollständig auffrischen

```