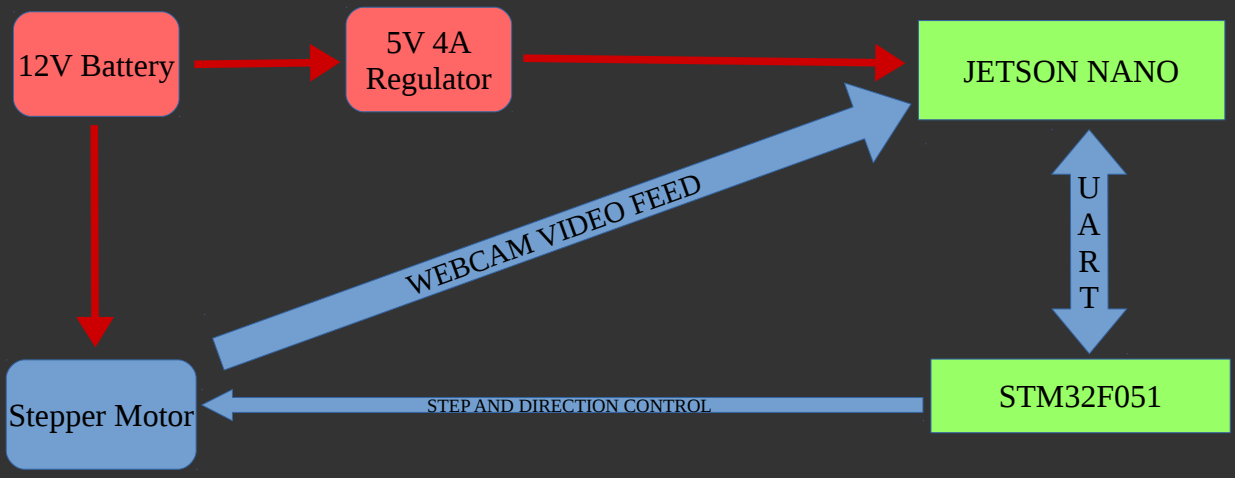
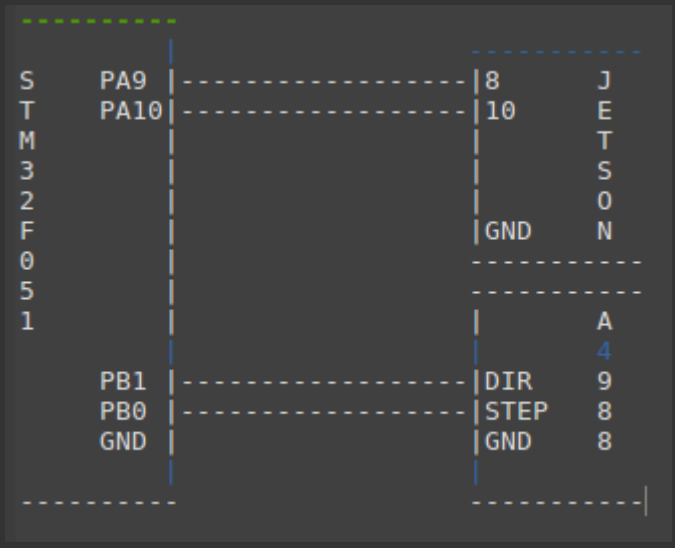
SYSTEM OVERVIEW:





MOTOR CONTROLLER: STM32F051

GLOBAL VARS:

```
#define width 320

volatile uint8_t DMA_BUFFER[6]; //6bytes of data to be sent via dma
volatile uint8_t SEND_BUFFER[6]; // 6 byte to be stored before sending to dma, updated every 90hz

volatile uint32_t microseconds; //unit is 2us not 1us

volatile bool status = 0; //used to keep step status
volatile bool isCalibrated = 0;
volatile bool dir = 0;

volatile uint16_t current_steps = 1200;
volatile uint16_t target_steps = 300; //max: 1200 current_steps == 180*

//as received from the uart
volatile uint16_t x_center;
volatile bool new_x_center_flag;

//will be used to keep the incoming uart data
#define INPUT_BUFFER_MAX_LENGTH 2
volatile uint8_t INPUT_BUFFER[INPUT_BUFFER_MAX_LENGTH]; //max 2 bytes
volatile uint8_t input_buffer_index = 0;

volatile bool time_calibrate_flag = 0; //used if the time zero button is pressed to reset the time on the next 90Hz sending interrupt
volatile uint32_t last_trigger_exti0_1 = 0; //used for debouncing the btn
volatile uint32_t last_trigger_exti2_3 = 0; //used for debouncing the btn
```

FUNCTIONS:

```
int main(void)
```

//initialisation functions

```
void init_TIM3(void);
void init_TIM14(void);
void init_TIM15(void);
void init_TIM16(void);
void init_TIM17(void);

void init_GPIOA(void);
void init_GPIOB(void);
void init_UART1(void);
void init_DMA(void);
```

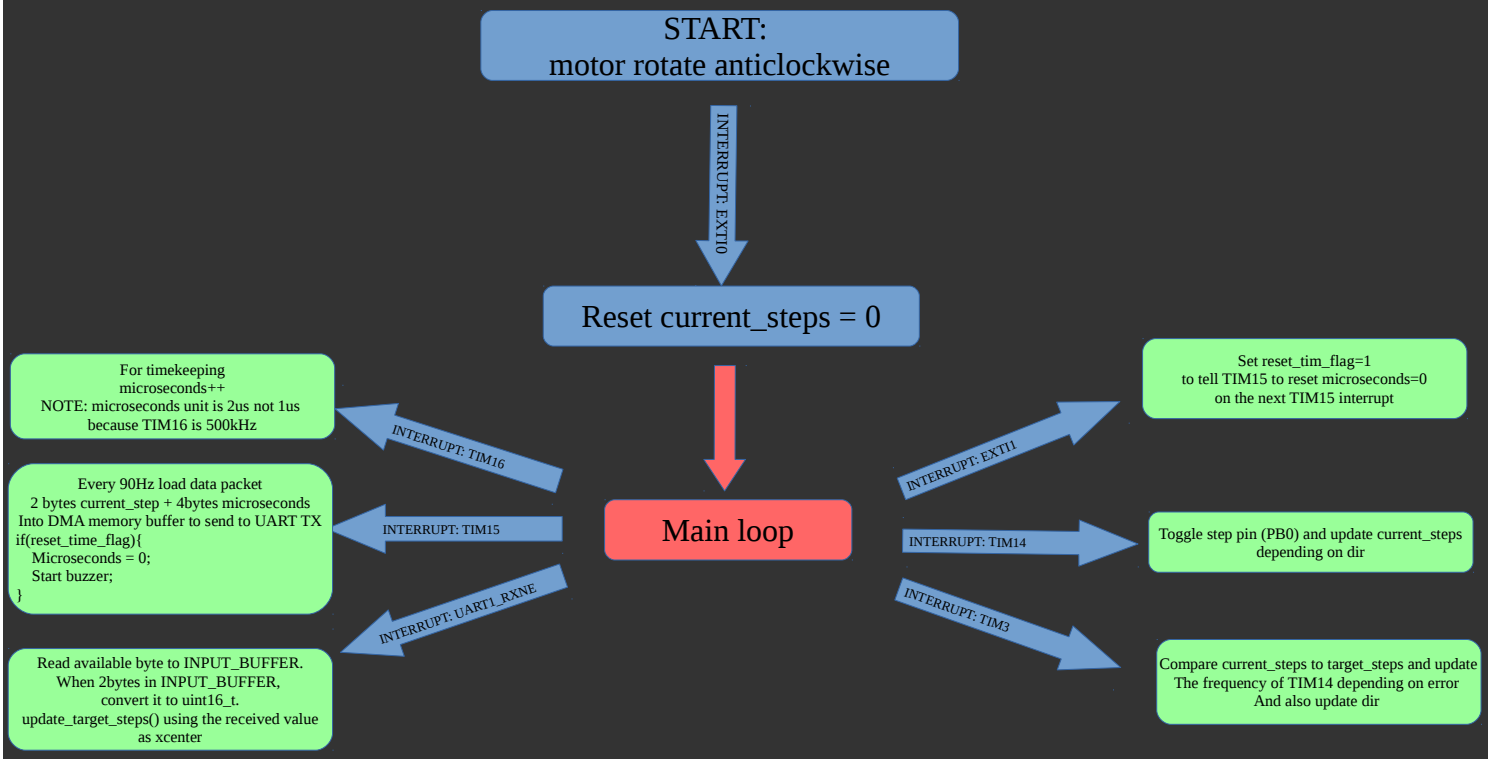
//auxiliary functions

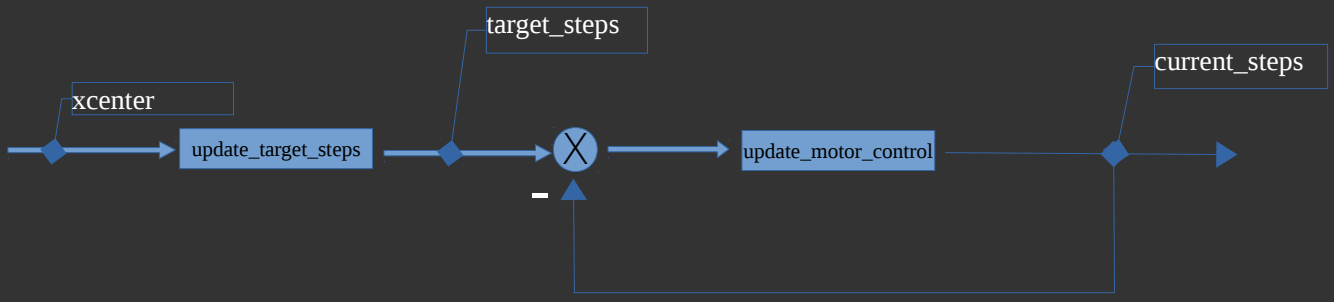
```
void update_motor_control(uint16_t current_steps);
void update_target_steps(uint16_t x_center);
void reset_time(void);
```

//Interrupt Handlers

```
void TIM14_IRQHandler(void) //frequency modulated for motor stepping
void TIM15_IRQHandler(void)// @90Hz send data to UART TX using DMA
void TIM16_IRQHandler(void) // @500kHz increment microseconds
void TIM17_IRQHandler(void) // one shot timer for buzzing
void TIM3_IRQHandler(void) // @32Hz update_motor_control
void EXTII0_1_IRQHandler(void)
void EXTII2_3_IRQHandler(void)//used unit step testing on PA1
void USART1_IRQHandler(void)
```

PROGRAM FLOW STM32F051:





OBJECT DETECTION: JETSON NANO

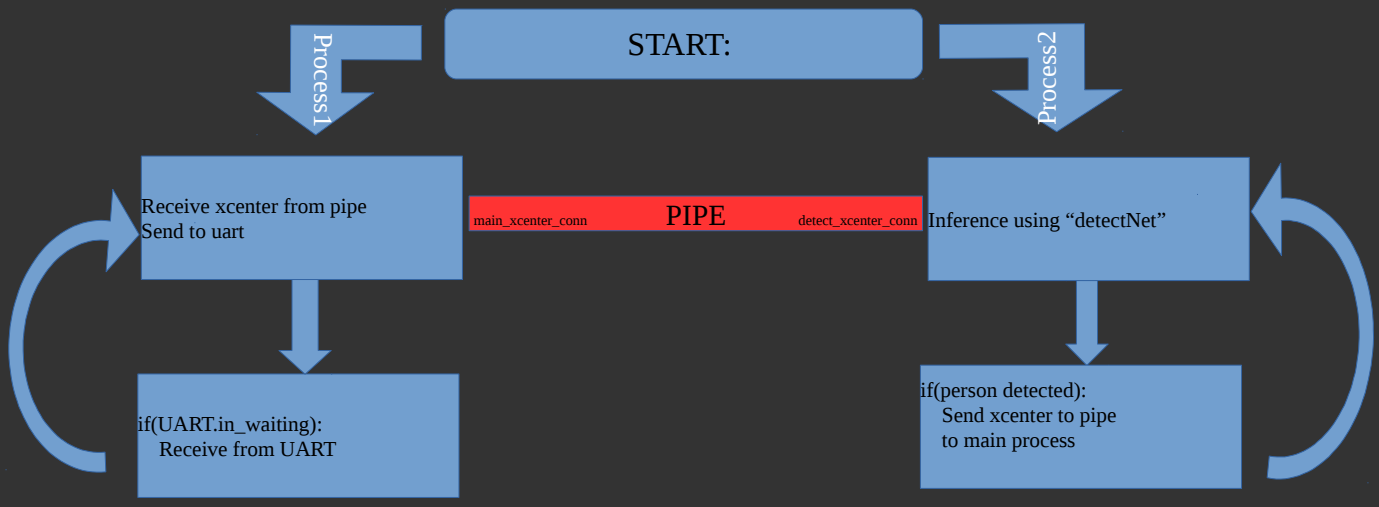
GLOBAL VARS:

microseconds = []
#holds all the timestamps received from uart.
#NOTE: each unit is 2 microseconds not 1 since the counter freq of the stm32 is 500kHz not 1MHz due to some issue I couldnt get the timer (TIM16) to 1MHz. total runtime before timestasmps overflow is ~ 240min

steps = []
holds the received steps receioved from thr uaurt at the correspomding timestamp

display_size = (320,240) #(width, height) 320, 240
uart = serial.Serial("/dev/ttyTHS1", baudrate=115200)

FUNCTIONS:

def object_detection(x_center_conn):
#will run on a seperate process and is using the NVIDIA detectNet inference engine to detect an object in the frame using the ssd-mobilenet-v2 network (API from https://github.com/dusty-nv/jetson-inference)

PROGRAM FLOW: