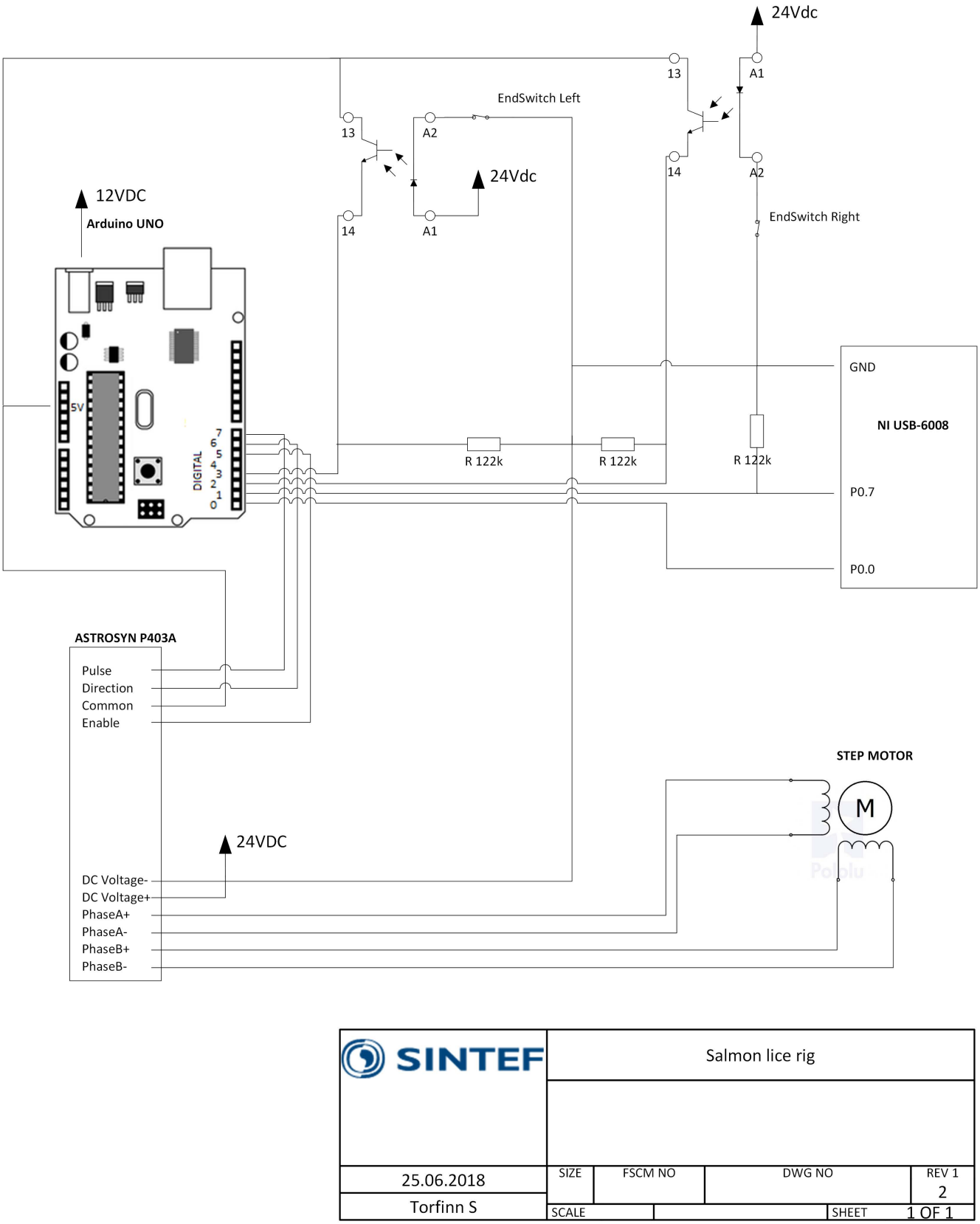
**Fig. S1. Motor control circuit schematic**



# Fig. S2. Arduino code

Programmer's Notepad - Arduino - StepMotor.iso

```
#define RIGHT     LOW
#define LEFT      HIGH

int Start_Move = 0;
int Pos_Reached = 1;
int EndSWRight = 2;
int EndSWLeft = 3;
int MOTORRESET = 4;
int MOTOREN   = 5;
int MOTORDIR  = 6;
int MOTORPULSE = 7;

int motorState = 0;
int microstep = 1;                                  // 1 - 250 via DIP-switch
int moves = 0;
long StepCounter;

void setup() {
  InitMotor();
}

void loop() {
  SetMotorDirection(RIGHT);

  while(1){
    if (digitalRead(Start_Move)){
      StepMotor(3300, 100);
      while (!digitalRead(Start_Move))
        delay(10);
      StepMotor(19200, 100);
      moves += 1;
    }

    if (moves == 5){                                // Last step
      if (digitalRead(Start_Move))
        StepMotor(3300, 100);
      moves += 1;
    }

    if (moves > 5){
      SetMotorDirection(LEFT);
      StepMotor(115800, 120);
      moves = 0;
      SetMotorDirection(RIGHT);
    }

    delay(50);
  }
}

void InitMotor(){
 // Astrosyn P403A driver, 24 - 40V dc
 // 0,9deg/step and microstep = 1 -> 400 steps / rotation
  pinMode(Start_Move, INPUT);
  pinMode(Pos_Reached, OUTPUT);
  pinMode(EndSWRight, INPUT);
  pinMode(EndSWLeft, INPUT);
  pinMode(MOTORRESET, INPUT);
  pinMode(MOTOREN, OUTPUT);
  pinMode(MOTORDIR, OUTPUT);
  pinMode(MOTORPULSE, OUTPUT);
  digitalWrite(MOTORDIR, HIGH);
  digitalWrite(MOTOREN, HIGH);
}

void MotorPulse(){
  if(digitalRead(EndSWLeft) || digitalRead(EndSWRight)) // Normally low - closes to 5V (via optocouplers) and end switch
    StepCounter = 0;
  else
    motorState = !motorState;

  digitalWrite(MOTORPULSE,motorState);
}

void SetMotorDirection(int Status){
  digitalWrite(MOTORDIR, Status);
}

void StepMotor(long steps, int delayMicro){
  digitalWrite(Pos_Reached, LOW);                   // Signal to LabVIEW - moving
  StepCounter = steps;
  long rampStep = 4;
  int RampDelay = 2500;  // Startdelay
  int RampA = RampDelay/rampStep;
  long StepLen = RampDelay;

  while((microstep*StepCounter) >= 0){
    MotorPulse();
    delayMicroseconds(StepLen);

    if (StepCounter <= RampA){                      // ramp down
        StepLen += (rampStep*1.1);
        if(StepCounter <= 0){
          digitalWrite(Pos_Reached, HIGH);          // Signal to LabVIEW - start taking images
          delay(50);
        }
    }
    else{                                           // Normal drive mode
      if (StepLen < delayMicro)
        StepLen = delayMicro;

      else if (StepCounter >= (steps - RampA))      // ramp up
        StepLen -= rampStep;
    }
    StepCounter--;
  }
}
```

## Fig. S3. LabView code

## FHF UL mod 2 eng.vi

The Low-Level Sequence VI acquires a sequence of images and displays the images in an image control using low-level acquisition VIs. A sequence is appropriate for applications in which you want to process multiple images.