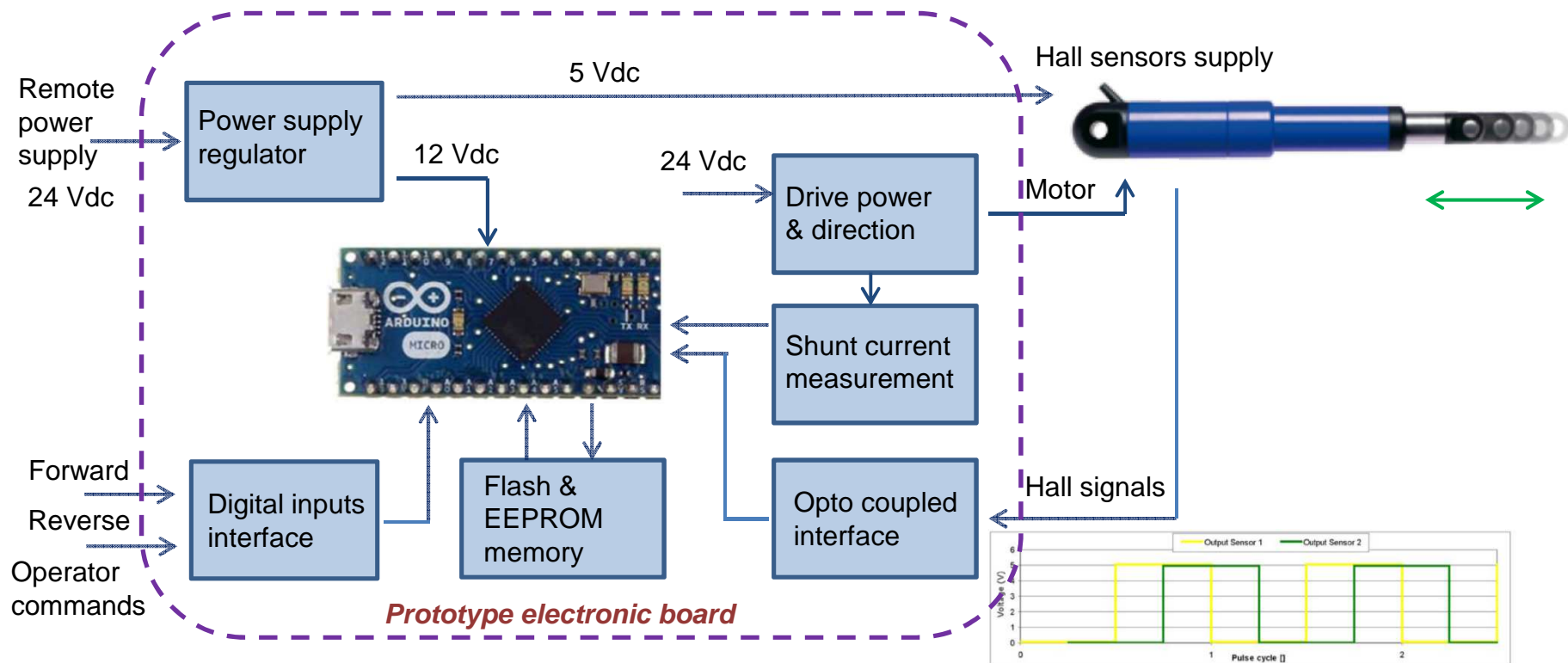


PART N° 2 Hardware description

Block diagram prototype electronic board.



"Linear actuator electronic control with ARDUINO" PART N°2

Requirements of prototype electronic board.

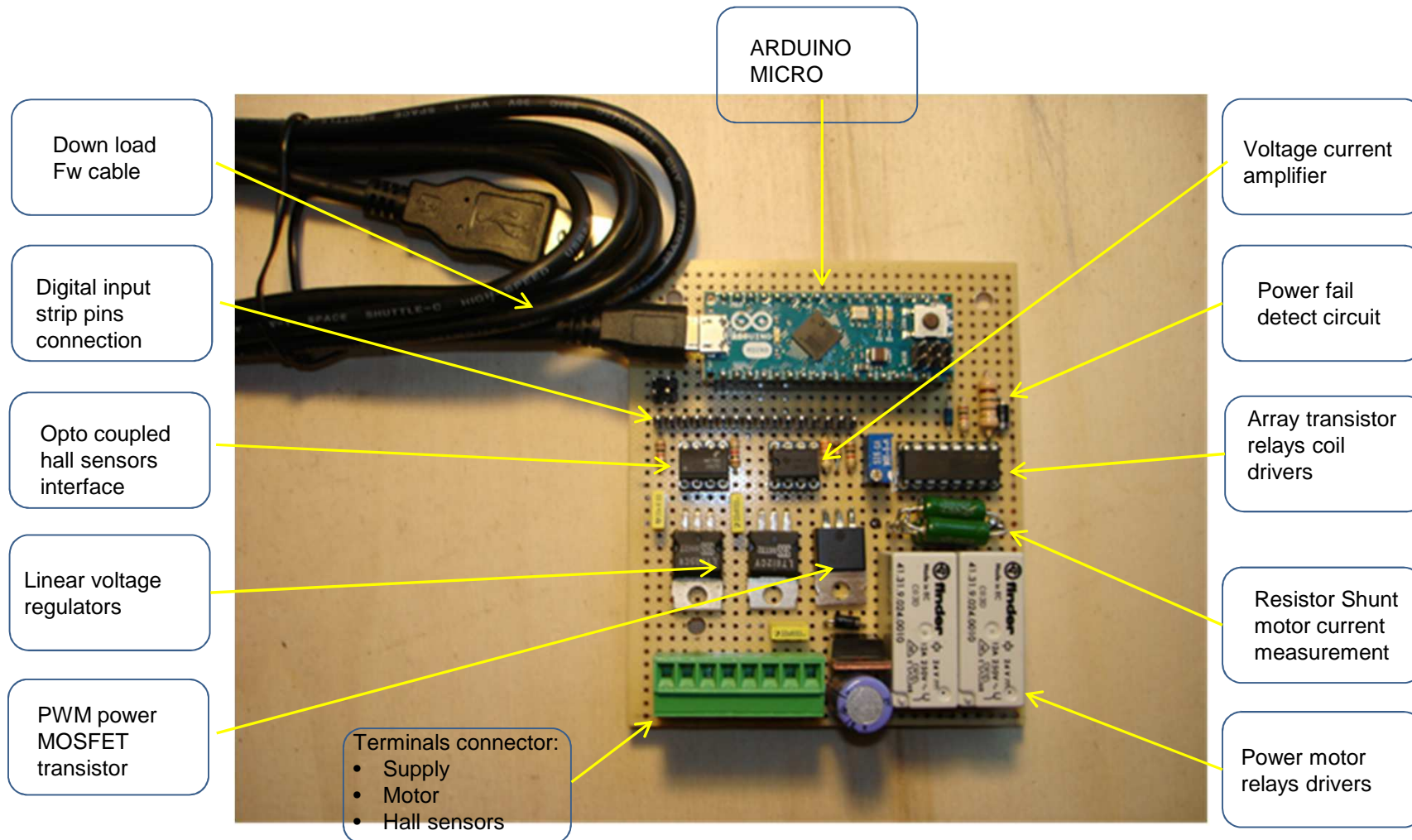
The main parts to control this actuator are:

- Power supply regulators.
- Drive power and direction of the motor.
- Counting rotary hall (encoder type) pulses and recognize of the direction.
- Recognize operator command Forward/Reverse.
- Control, max current and motor power dissipation.
- Saving and restoring the count position at power off/on.

Starting by these, other functions may be added:

- Generator ramp start/stop.
- Zero point calibration sequence.
- Programming and saving of the SW limit positions.
- Trapezoidal speed profile generator.
- Electronic gearing mode
- Motion tasks sequencer

Servosystem prototype ARDUINO MICRO based



"Linear actuator electronic control with ARDUINO" PART N°2

Main electrical features of prototype electronic board.

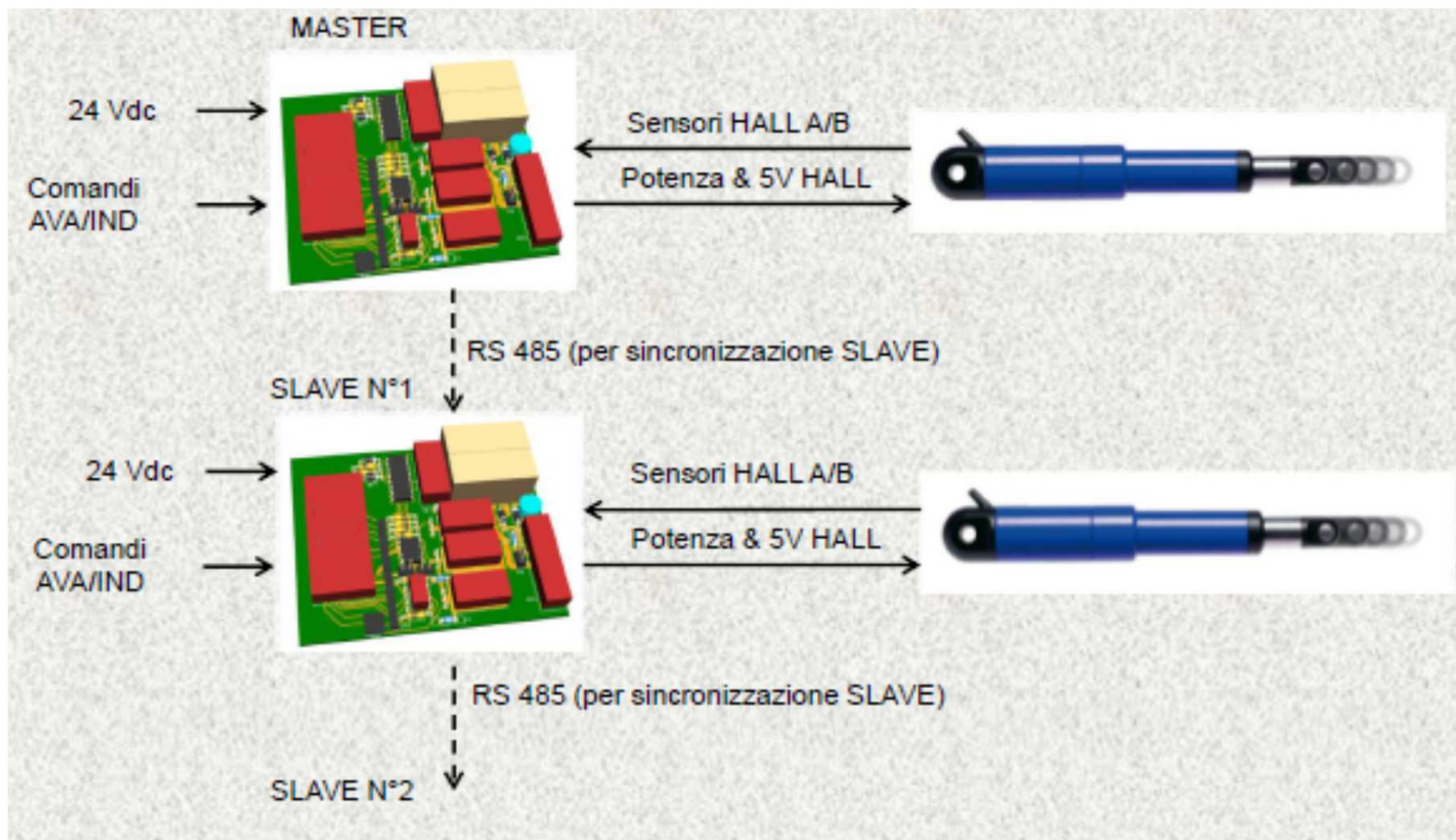
The picture above shows the prototype developed in part with recovered components. In particular aren't been optimized the power supply and the circuits of commands relay. In the case of an engineered work, the suggestions are:

- Use of SMD components
- Input power supply based on a switching regulator.
- Substitution of ARDUINO MICRO board with the equivalent ATMEL microprocessor. In the ARDUINO IDE environment there are instructions to load your program into a ATMEL chip with cleared memory.

The main electrical features of the prototype board realized , are:

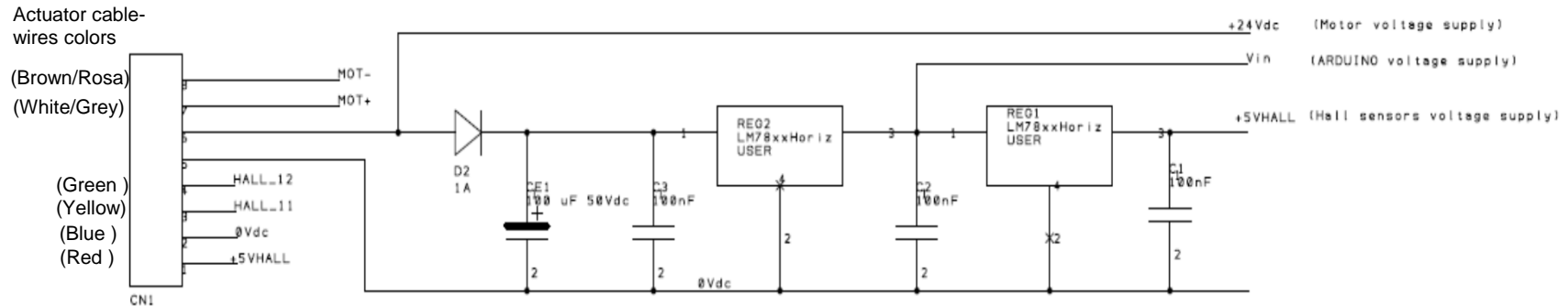
- Controller based on ARDUINO MICRO
- Motor command based on two relays and one power MOSFET driven with PWM
- Shunt resistor for motor current measure.
- Hall sensors interface, opto coupled.
- Expansion serial port TX/RX and digital I/O for "gearing mode" with other boards and actuators (Multi control synchronized actuators).

Schematic representation of a multi control synchronized actuators.



"Linear actuator electronic control with ARDUINO" PART N°2

Power supply circuit

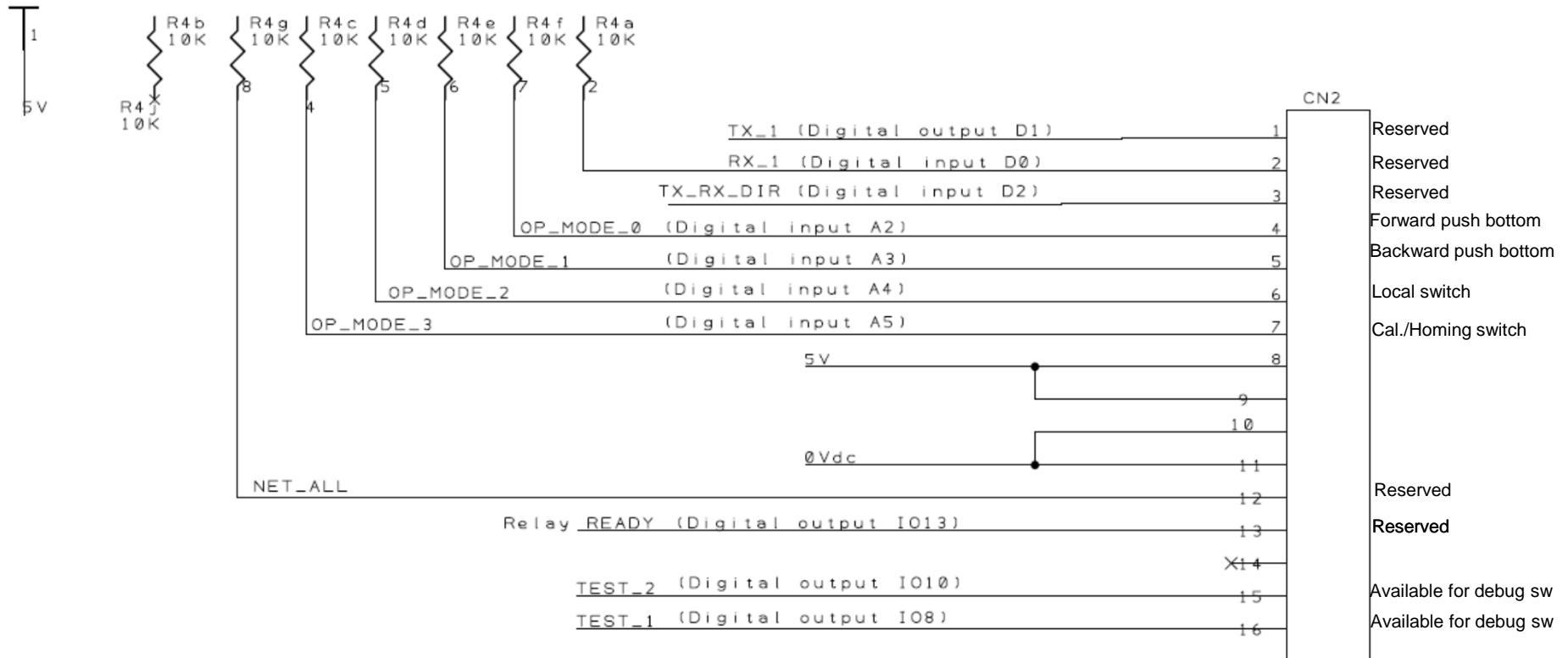


The 24 Vdc is the supply required by the motor of the actuator. This is the main power voltage supply for the electronic board. This voltage is also monitored by the POWER FAIL circuit, described more forward.

For simplicity, on this prototype are been used two step down linear regulators. One from 24 to 12 vdc for ARDUINO MICRO and another from 12 to 5 Vdc for the voltage supply of hall sensors. The disadvantage of this circuit is the power dissipation into the linear regulators due to the voltage difference between input to output multiplied for the flowing current.

For the prototype this is acceptable. For an engineered board, the switching voltage regulator is the best solution. Note that it is possible to supply ARDUINO MICRO and the hall sensor devices with a single 5 Vdc.

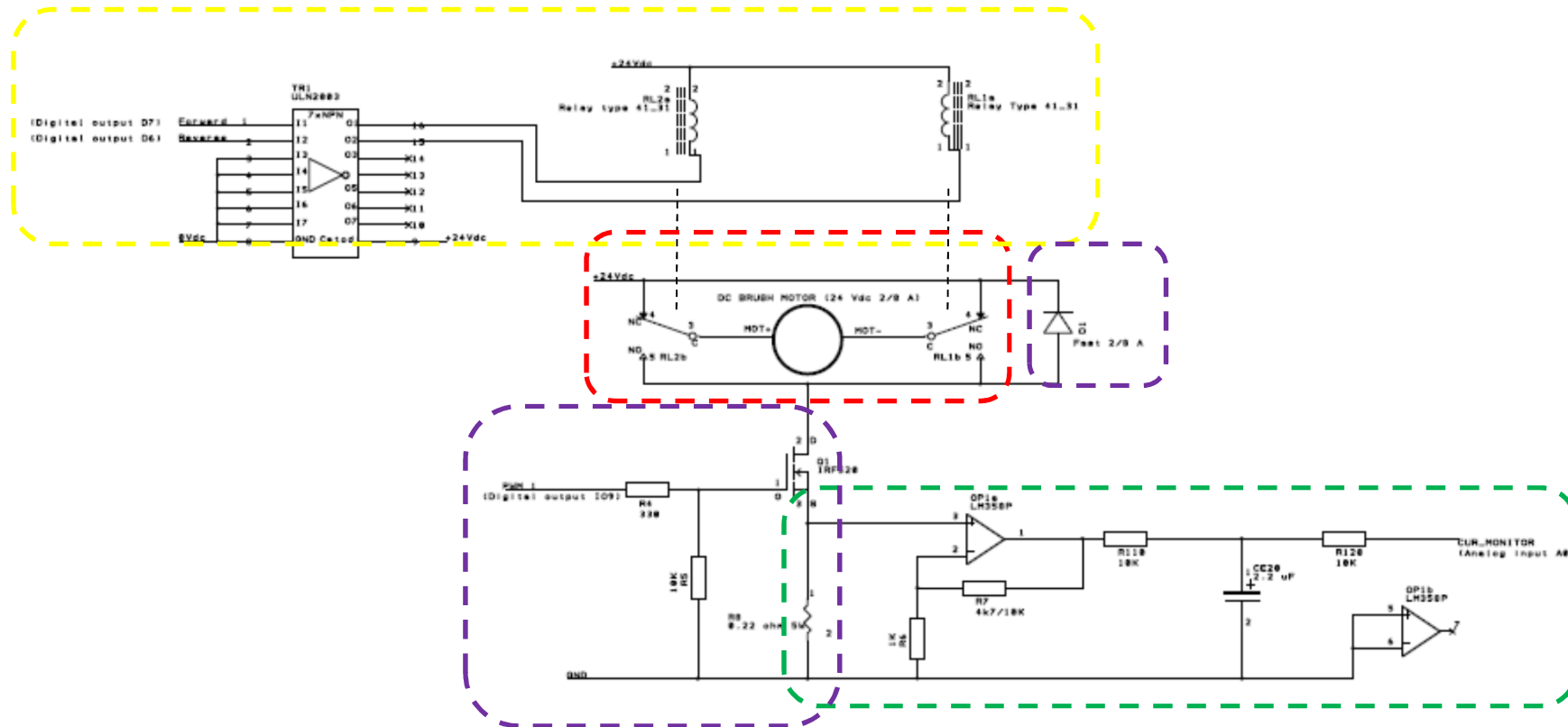
Digital input strip pins connection



"Linear actuator electronic control with ARDUINO" PART N°2

10071 Borgaro T.se (TO) - Via Mappano, 17 - T +39 011 451 8638 r.a. - F +39 011 470 4891

Drive motor power circuit (Schematic diagram)



"Linear actuator electronic control with ARDUINO" PART N°2

Drive motor power circuit (description)



The power drive circuit has four parts:

- The interface of ARDUINO to command the coils of the relays (Yellow area). Has been used an array of transistors with diode recovery with common cathodes connected to 24Vdc power supply.
- The two relays are with single switch and contacts by 12 ampere (NO and NC) (Red area). The circuit normally holds the motor phases in short circuit and it brakes external attempts to motor rotation. When is active one of the two relays and the PWM circuit is on, the current flows in the motor. The current direction depends by the relay active. The direction of the current inside the motor drives the revolution of the motor clockwise or counter clockwise.
- The PWM power circuit (Violet area).
- The converter and amplifier current to voltage (Green area). The gain of the amplifier is about 5. Due to the PWM control, the shape of the current is pulsed. The filter circuit in cascade to the amplifier transforms the pulses in a rms value. The constant time of the filter on this prototype has been set to about 20 ms.

Summary of signals inputs and outputs of ARDUINO.

This circuit is connected to the following ARDUINO MICRO signals:

- D7 as digital output to command Forward relay
- D6 as digital output to command Reverse relay
- IO9 as analogue output, to command the PWM1 signal, of the power MOSFET transistor
- A0 as analogue input to measure voltage signal current measurement.

This section is the most critical part of the project. It is a compromise between the performances requested to the electronic card and the processing resources of ARDUINO MICRO.

The first requirement is the capability of counting the pulses at maximum speed of the motor revolutions.

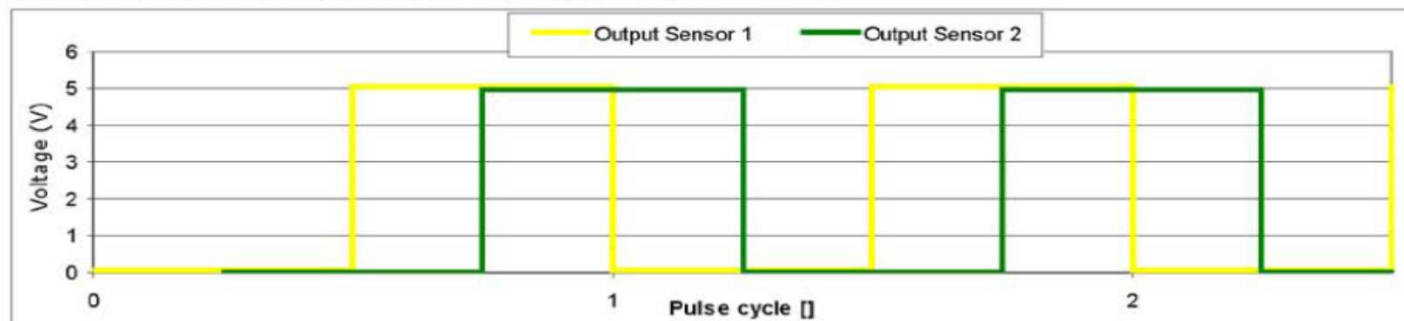
To do this you must consider that each sensor generates a pulse for revolution of the motor.

The motor of the CONCENS actuators, without load may reach about 8000/9000 RPM.

This means $9000 / 60 = 150$ pulses per second for each of the two sensor. One every 6,6 ms.

The pulses generated by the two sensors are shifted of 90 degree.

Hall signal output yellow and green cable. $\frac{1}{4}$ cycle delay between output sensor 1 and 2.
Order depends on displacement direction of the actuator.



Excluding a hardware counter circuit, remain two SW approaches to solve this requirement:

- Polled input mode
- Interrupt input mode

Interface and counting pulses hall sensor circuit (2 continue)



The polled mode requires all speed performance of ARDUINO focused on this task, with limited resource to do other tasks. The mode interrupt provides a very efficient method, but requires two inputs with interrupt reserved.

In this project the suggestion is a compromise of the two methods that use standard functions available in the ARDUINO's IDE environment.

The method consists in a routine, under control interrupt, attached to the interrupt level reserved of the TIMER1 function (standard is one interrupt about every 2 ms).

To do this you must include the library "TimerOne.h" following the instructions that you can find on ARDUINO IDE environment.

With this library it is possible also to modify the period of interrupt interval time of TIMER 1, to respect the minimum sample time of hall signals at maximum motor speed.

In this prototype the period has been set to 500 μ s (2 kHz) and the hall sensor signals are wired to two normal digital inputs (IO12 and IO11).

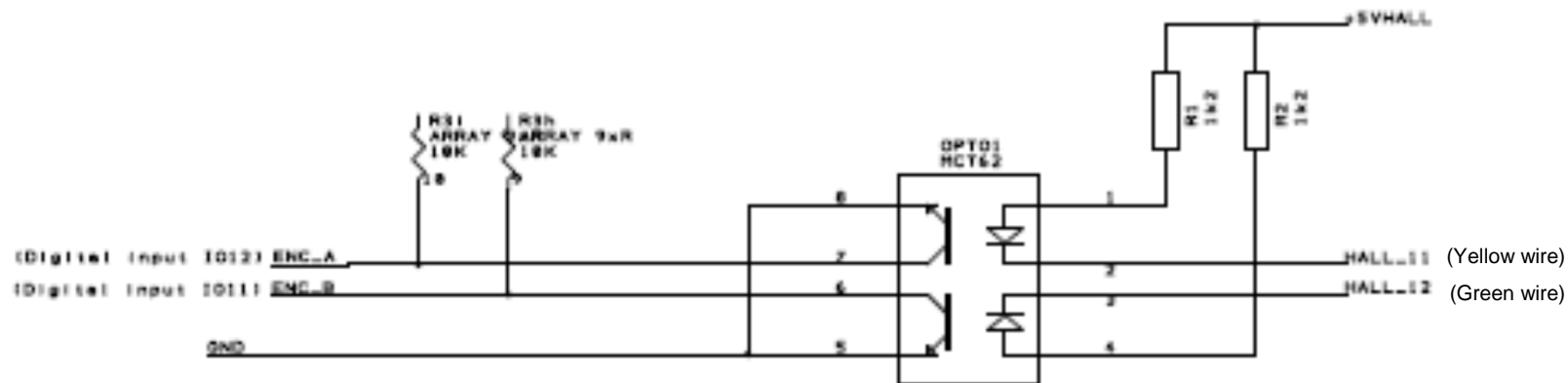
The routine requires two consecutive samples to decode the pulse transitions of the channel A or of the channel B, thus for a complete motor revolution are necessary at least 8 samples. The max hall sensors frequency must be less of:

$$8 * 500 \mu\text{s} = 4000 \mu\text{s} = 250 \text{ Hz}$$

that is greater than 150 Hz of the specifications.

Be careful that the signals may have a jitter. With interrupt at 500 μ s, consider 150 Hz max permissible hall sensors frequency to have margin and to avoid pulse lost transitions.

This modification also influences the standard PWM1 output frequency from 500 Hz to 2 kHz, but this is better from motor coil point of view.



The circuit uses a double opto isolator. This permit different voltage supply of the sensors hall, for example the same +12 V of the ARDUINO MICRO, that could to permit (modifying proportionally the value of the series resistor) more long cable between CONCENS actuator and the electronic board.

Be careful that the CONCENS specifications limit at +5 Vdc the max voltage supply for the HALL SENSOR,
But +12 Vdc should be accepted limiting sink current at 5 mA.

Count and multiplication x 4 of the Hall sensor pulse



The developed sw routine when woken up by the interrupt, tests the logic level of the two inputs that interface hall signals, and together to the same signals tested previously (during the interrupt before) forms a binary variable of four bits. By this for each transition state (count x 4) is possible decodes the count pulses, UP and DOWN.

At each interrupt the routine read INPUT and decodes the following integer variables.

Variable	Description	Bit position in the binary variable
• CHA	(actual level of the digital input ENC_A)	Bit 0
• CHA (-1)	(state of CHA at previously interrupt sample)	Bit 1
• CHB	(actual level of the digital input ENC_B)	Bit 2
• CHB (-1)	(state of CHB at previously interrupt sample)	Bit 3

The values may be only 0 or 1

With a single numerical operation is possible evaluate a code by which to recognize if it has been a count transition and if it must to be count up or down. The numerical operation is

$$\text{ENCODER1_dec} = \text{CH1A} + \text{CH1A} (-1) * 2 + \text{CH1B} * 4 + \text{CH1B} (-1) * 8$$

The ENCODER1_dec is an integer variable that can assume different results:

a) The routine increments the sw counter, when the result is one of these:

2, 4, 11, 13.

b) The routine decrements the sw counter, when the result is one of these:

1, 7, 8, 14

The other results are ignored and the counter is not modify.

Timing count pulses with change direction

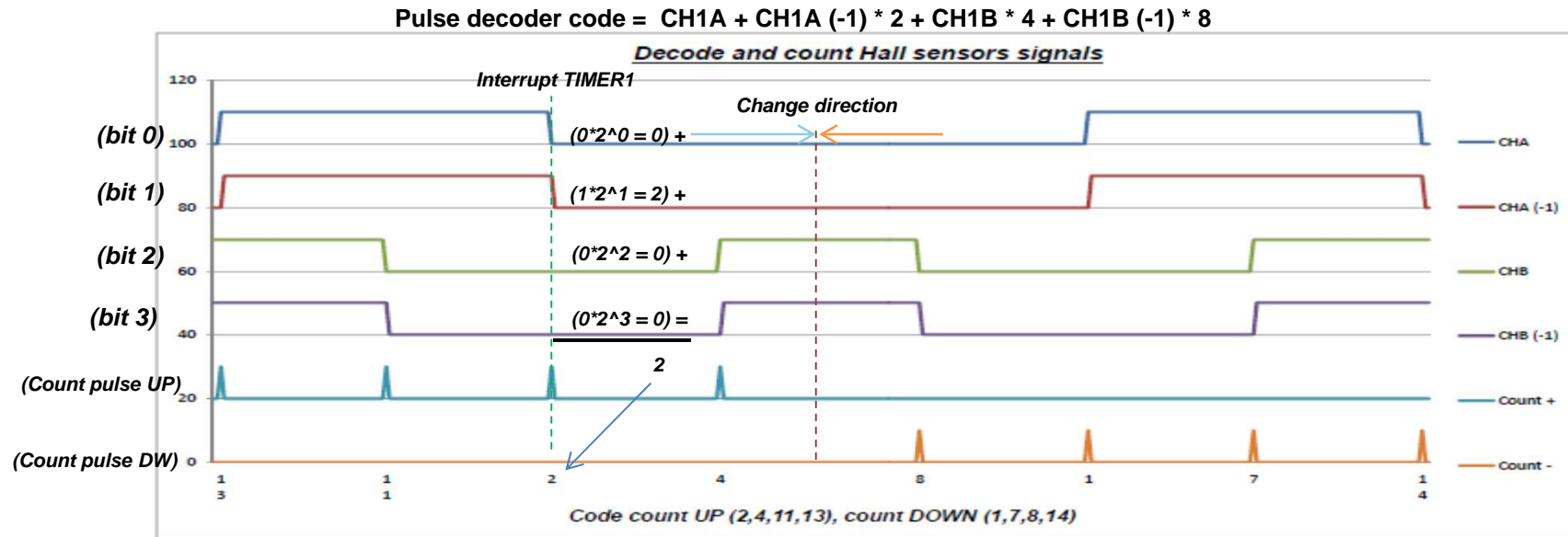
The following diagram timing show the decode logic.

Assuming that the positions of the four variables CHA, CHA (-1), CHB, CHB (-1) form a binary variable of four bit:

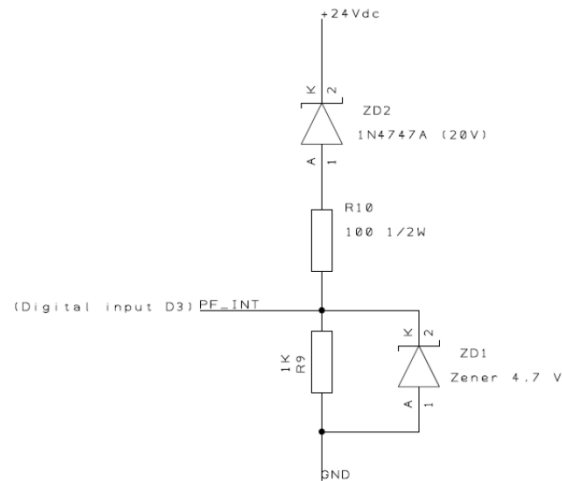
- | | | |
|--------------------------------|-------|---|
| a) CHA is less significant | bit 0 | (value equal at state level $(0/1) * 2^0$) |
| b) CHA (-1) | bit 1 | (value equal at state level $(0/1) * 2^1$) |
| c) CHB | bit 2 | (value equal at state level $(0/1) * 2^2$) |
| d) CHB is most significant bit | bit 3 | (value equal at state level $(0/1) * 2^3$) |

Then the result of the numerical operation:

identify the codes of the valid transitions and the direction of the movement

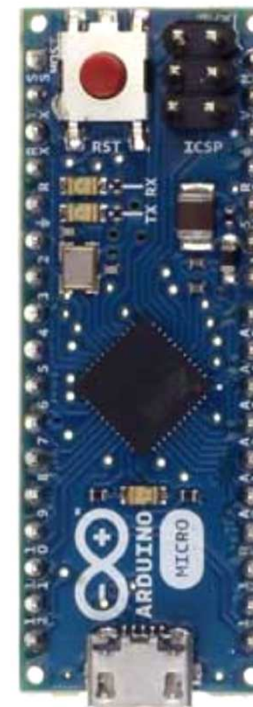
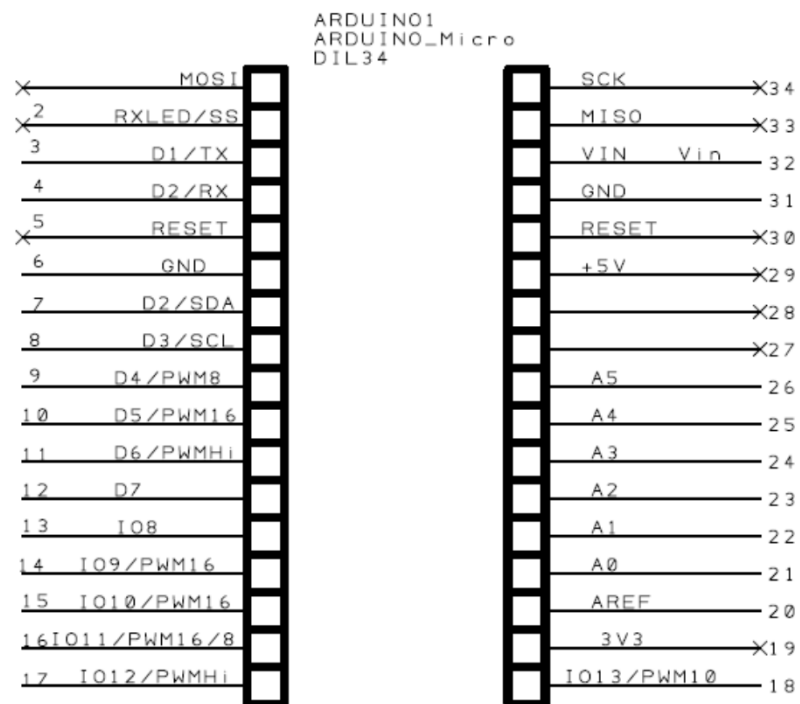


Power fail detect circuit



This type of actuator requires an absolute position information at power on. The Hall sensors give only an incremental position information. The problem can be solved saving in a EEPROM the actual position during power down and restoring it at power up. The linear actuators CONCENS have an ACME screw that are not reversible. During the power off the actuator holds the position. The circuit above is used to generate a signal when the external power supply goes away. The ARDUINO and HALL sensors are functioning until the +24 Vdc remaining over the 14 Vdc

Signals allocation of the ARDUINO MICRO



Next step “SW description” (“Linear actuator electronic control with ARDUINO” PART N°3)

“Linear actuator electronic control with ARDUINO” PART N°2