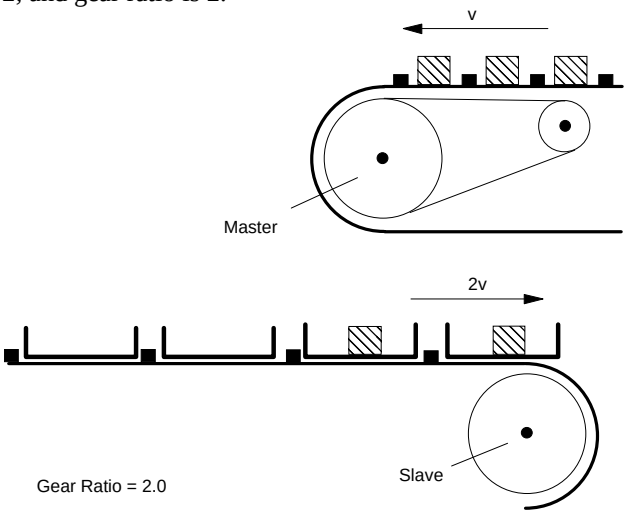
5 Electronic Gearing Programs

The four applications that will be covered in this section include:

1) Single gear ratio motion program
2) Variable gear ratio motion program
3) Engage in electronic gearing when external signal changes state
4) Engage in electronic gearing when master passes a programmed position

Illustrated below is an example of a packaging process that includes two conveyor belts. The upper belt contains the products equally positioned in between the logs. The master motor moves the product and drops each into the buckets. Clearly, this calls for a gearing mechanism that engages the master and slave, the conveyor belt moving the buckets. The gear ratio in this example is determined by the ratio of the space between the centers of adjacent buckets and the space between the products. In the following example, the motion program runs only one master/slave line. This line states master is motor 1, slave is motor 2, and gear ratio is 2.



1) Single gear ratio motion program

```
#include    “init_mx4.hll”

plc_program:
    run_m_program(electronic_gearing)
end


electronic_gearing:
    gear(0x1, 0x2, 2)
end
```

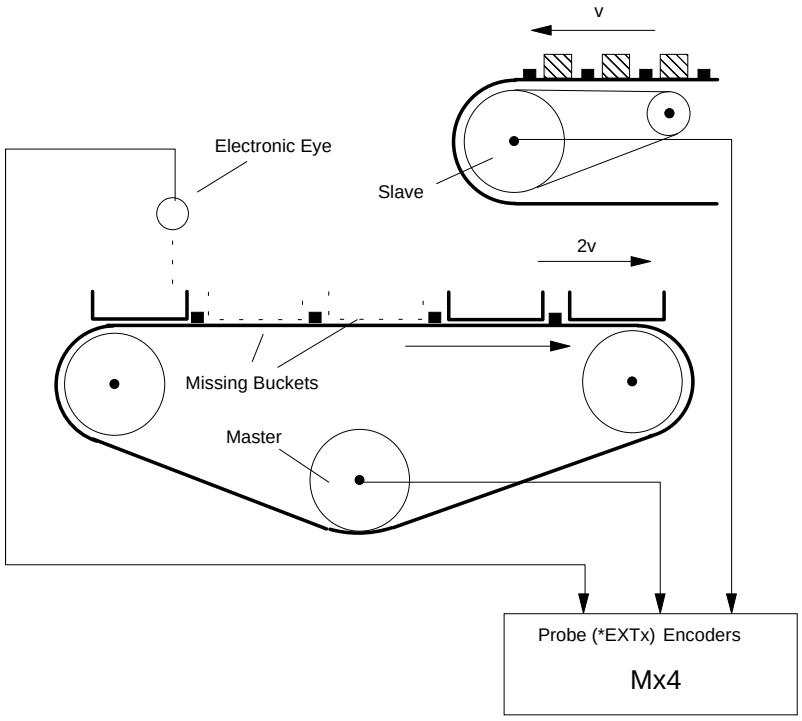
2) Variable gear ratio motion program

In this example, motion program `electronic_gearing` starts an endless loop in which variable `gear_ratio (VAR4)` is continually updated. You may use the second task (permitted in DSPL programming) to calculate `gear_ratios` on-the-fly. Alternatively, if the host is to update `gear_ratios`, the host based real time command `CHANGE_VAR` (contained in Mx4 C++ or Visual Basic DLL) can be used to update `VAR4`.

```
#define    master     0x1
#define    slave      0x2
#define    gear_ratio var4

#include          “init_mx4.hll”

plc_program:
    run_m_program(electronic_gearing)
end


electronic_gearing:

    gear_ratio = 2

    while (var1 == 1)     ;changing var1 (by host) disengages slave
           gear(master, slave, gear_ratio)
           delay(100)
    wend

    gear_off_acc(2)
end
```

3) Engage in electronic gearing by an external signal

In this example, the slave is geared to the master motor only if the pulse sent by the electronic eye is switched to logic zero. This feature is useful in applications where there may be a problem on the line such as missing bucket.

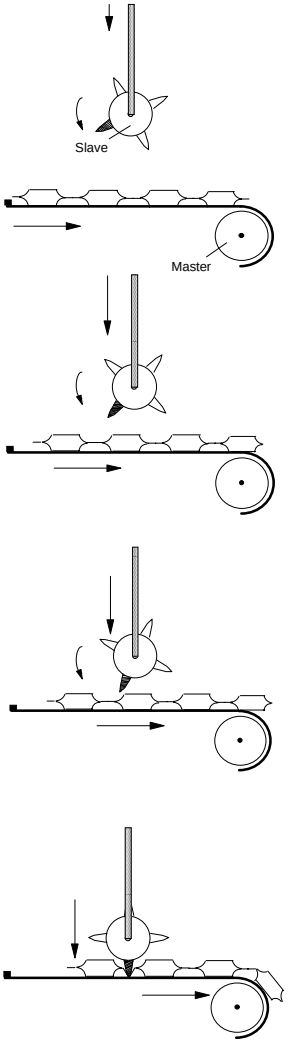



```
#define    master     0x1
#define    slave      0x2
#define    gear_ratio var4
#include    “init_mx4.hll”

plc_program:
    run_m_program(electronic_gearing)
end
```

```
electronic_gearing:

    velmode (1,5)                ;put master in velocity control mode

    gear_ratio = 2

    gear_probe(master, slave, 1, gear_ratio)

    wait_until(INP1_REG & 0x0002) ;wait until stop button is pushed
    gear_off_acc(2)

end
```

4) Engage in electronic gearing when master passes a programmed position

Products on the conveyor belt moved by the master motor are positioned uniformly. The slave motor cuts the film connecting the two adjacent products. The result of this cut is unsatisfactory if the knife lands vertically. It is preferred that while landing, the knife-edge travels and is tightly geared to the position of film that must be cut. This is shown in the following figure.



```
#define    master     0x1
#define    slave      0x2
#define    gear_ratio var2
#include    “init_mx4.hll”

plc_program:

    run_m_program(electronic_gearing)

end



electronic_gearing:

    gear_ratio = 1

    gear_pos(master, slave, gear_ratio, 200) ;engage when master passes 200
    velmode (1,5)                            ;start master move
    wait_until(INP1_REG & 0x0002)            ;wait for stop button
    gear_off_acc(2)                          ;stop slave
    stop(1)                                  ;stop master

end
```