PHASE
MOTION
CONTROL

BOLLETTINO TECNICO N. 003/97- TECHNICAL BULLETIN N. 003/97

FROM TRANSMISSION SHAFTS TO ELECTRONIC CAMS: HOW TO COORDINATE MULTIPLE AXES IN AUTOMATIC EQUIPMENT

Most large automatic machines need to synchronize various operations, e.g. when the workpiece passes from one station to the next, or in continuous processing. Traditionally, this is performed by using a single, large prime mover, such a large motor drive. Motion is then distributed with a transmission shaft, or a chain, or a set of gears. This approach can become critical for large and accurate machines, such as printing machines, and is anyway inflexible in terms of process programming, as a variation in the motion profile of each operation depends on a change of a piece of transmission or a cam. The current trend towards greater flexibility and programmability, dictates the use of distributed servo motors, each performing a local operation according to a remotely programmable motion profile, each synchronized to the other stations, as if they were connected by a shaft. In other words, a generalized “electric shaft”.

There is no common consensus, at present, on how to resolve this general application problem in the most effective way. If the synchronization need is limited to two shafts, the function can be performed by digital servo drives which offer the “electric shaft” function; however, if the axes are several, and the motion profiles are synchronized but not equal, a central controller with coordination and monitoring functions is unavoidable. Such controller could be a NC, a PLC or an industrial PC.

The first technical choice is the following:

- Either the central controller is entrusted with the generation of the position references only, leaving to the drives the execution and verification of the motions;
- Or the central controller closes a feedback loop on the actuations, generating the profiles and receiving the position sensor signals, and feeding the dives with either a speed or, more elegantly, a current or acceleration reference.

The limits of the first approach are due to the lack of a standard protocol for the transmission of an absolute position reference and of a proper diagnostic signal; the only available standard, incremental and open loop, was established for stepper motors. Consequently, this approach is reserved to those processes where a loss of synchronization is not critical, and where stepper motors can be safely used, i.e. with relatively slow dynamics; or to closed package systems, in which the controller and the NC are lumped in a custom unit and data are exchanged on the internal, parallel bus.

The second method is intrinsically open, and carries the most evolution potential. With this system, the traditional architecture is the following:

1 The controller has an adequate number of single or multiple axis control cards, which receive the encoder signals, and create a speed reference;
1 The power drive is fed a speed reference from the controller, and operates the motors accordingly

This approach performs well, but has a basic drawback: the position loop is on the controller side, the speed loop on the drive side. There are historical reasons for that, but in fact both loops require the encoder signal, and both are critically influenced by the mechanical parameters of the system. As a consequence:

1. Each drive needs to be programmed differently for each axis, according to the mechanical properties of the axis;
2. The encoder signal needs to be duplicated;
3. The speed reference is noise and offset sensitive, needs a high dynamic range and a wide bandwidth, making it rather critical both in analogue (differential signal) or digital form (16 or more bits?)
4. Advanced control algorithms, such as adaptive control or system identification, are impossible as the system knowledge cannot access the parameters stored in the individual drives.
5. There is a frequent conflict between the space loop on the controller and the speed, or space derivative, loop on the drive resulting in noise or limit cycling

These limitations can be overcome, at least in part, in two ways:

1. either saving the traditional architecture but upgrading the data transmission from analogue to digital, be means of a very fast field bus;

2. or moving the speed loop on the central controller and programming the drives as simple current or acceleration controlled drives.

The first solution does not require any substantial innovation, but purely a large deployment of distributed intelligence on the drives, needs a fast field bus (at least 10 Mbaud) and is therefore quite heavy from an hardware viewpoint. It requires a digital drive, usually with an additional communications board. The fast bus carries both the drive programming data at start-up, and the speed reference as well as the encoder data, which is needed both by the drive and by the central controller, during operation. This solution performs well, in spite of being quite heavy from an hardware and cost viewpoint. At the moment, the only bus which is fast enough for this operation is SERCOS, which is anyway bandwidth limited to 4-8 axes at most. System identification, parameter adaptation and active control remain impossible.

The second solution is more innovative, radical and elegant. The speed loop, which is just derived from the position loop, is transferred on the central controller axis boards. The reference to the drives is just a current reference or acceleration reference, which have no parameters depending on the system mechanics. This way, all drives are equal and interchangeable and need no programming at all. Additionally, a current reference is considerably less noise sensitive than a speed reference, is offset insensitive and requires a lower dynamic range (12 bits are more than adequate). On top of this, all the system parameters are inside the central controller so that system identification, adaptive control and innovative control algorithms can all be implemented.

The down side of this solution is the additional processing speed requirement which is placed on the central controller (PC, PLC or NC) axis processor or controller boards. In practice, if the speed reference processing is downloaded on the drive, a axis processing cycle of 500 μsec or even 1 msec is generally adequate; if a current reference is output, a similar shaft control quality is obtained with cycle times as short as 100-200 μsec. In the past, this speed limit prevented the diffusion of this simple, elegant and inexpensive technique. Present day controllers, however, are breaking the speed barrier without significant cost penalties.

With the current reference technique, it is possible to implement a synchronization of several axes in with a simple industrial PC equipped with multiple axes boards, efficiently and with good economics, using a set of current controlled drives which are interchangeable and not programmed.

In principle, almost all drives on the market are suitable for current control; in practice, as current control is the simplest form of control, the best drives for this applications should be the simplest; in fact, “intelligent” digital drives for this application are an useless waste of unnecessary (and unused) computing resources. The ideal current fed drive should be just simple, efficient and fast.

Just for this type of application, Phase Motion Control developed the AX^4 range of single board IGBT drives, which are based on a single ASIC and embody a minimal component count. Furthermore, the AX^4 drives have a current control loop which is inherently stable and 100 μsec delay time h for all motors of the ULTRACT II series.