# Design of an M6800 Based

By Mike McCormick
MCM Systems
247 E. Grove Street
Oneida, NY 13421

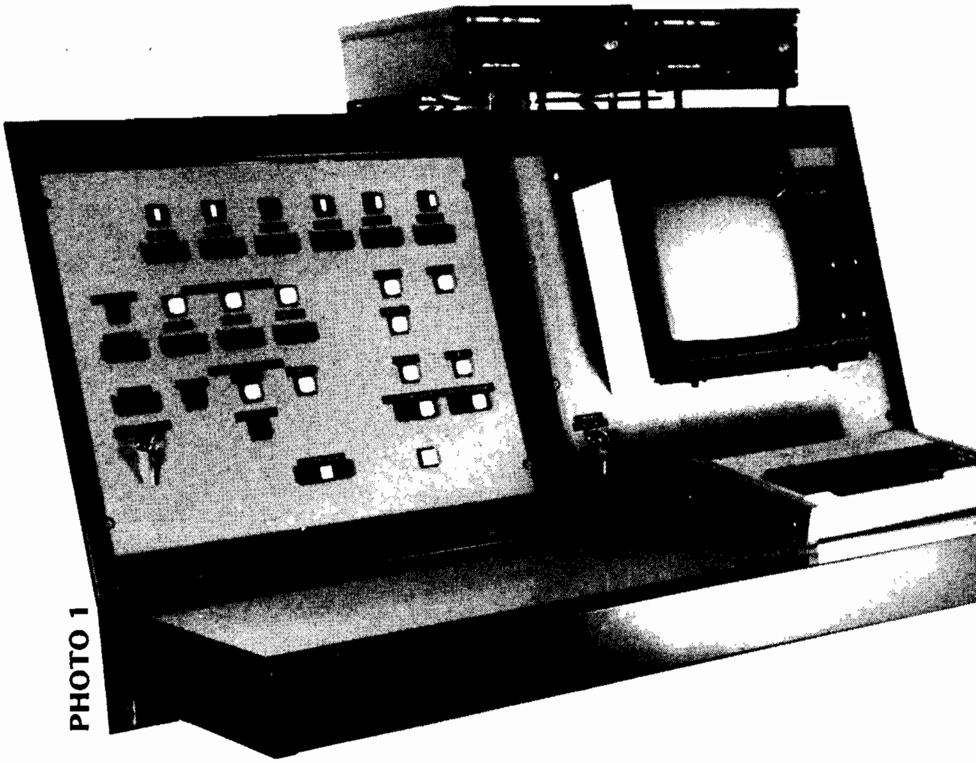

PHOTO 1

## INTRODUCTION

The M-6800 has been successfully used as a real time process controller for automatic weigh batching systems. The weigh batchers are marketed by MCM Systems to ready mix cement producers and are also rapidly gaining acceptance in other applications which require precision batching of dry material and fluids for operator selected mix designs.

The MCM Systems unit, illustrated in Figure 1, is an integrated process control system consisting of a manual control panel and a process control computer with appropriate interfaces to material batching control and measurement circuits. In this application the manual control panel provides full manual back up control for all material batching processes. However, the normal mode of operation is full automation under control of the processor.

This article describes the design and optimization of an M-6800 based controller for the system. Following a brief

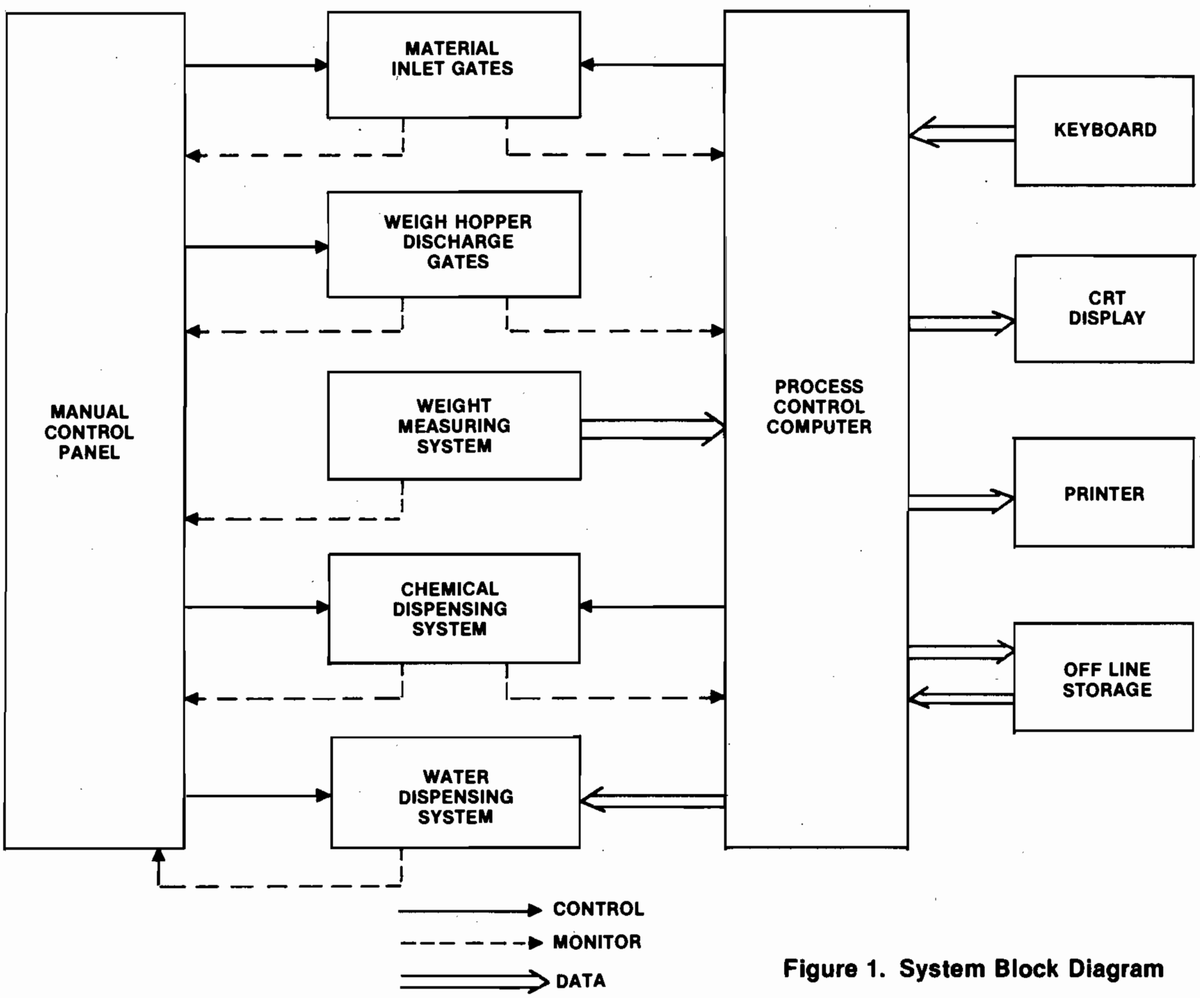


**Figure 1. System Block Diagram**

# utomatic Weigh Batching System

erview of the batching process the hardware and software sign processes are reviewed.

## ATERIAL BATCHING REQUIREMENTS

Precision batching of materials within a cement plant re- ire a sequence of mechanical gate openings and closings the inlet of aggregate and cement materials into central igh hoppers in accordance with a prescribed mix design. so, when required for the mix, air entraining fluids (admix- es) are batched into holding vials for subsequent dis- arge into the cement. The controller which automates ese processes has been designed to meet cement industry tching standards. This necessitates that within a given tching cycle, material batch weights must be held within ecified tolerances. If an out of tolerance condition occurs thin a batching cycle the controller must stop the cycle, k out machine controls and await the manual intervention restart the batch, following correction of the error condition.

## ONTROL PROCESSOR REQUIREMENTS

The controller for the batching system is the master execu- e for the weigh batching process. It controls all mix design ta input, schedules the batching and handles batch recycl- g. It also detects batching errors and external plant failures.

At the beginning of a process, the system accepts operator uts of material batch weights. These may either be meric keyins in response to processor generated prompts standard mix design material parameters which are recalled m permanent storage. Following the data input mode it mputes all batching tolerances, displays all material rameters and prompts the operator to start the cycle.

In response to the operator initiated start code, the system executes the batch cycle. This consists of a sequential opening and closing of various material inlet gates. In each case the close command is generated on the basis of accumulated weight during the material inlet process.

At the completion of the batch cycle the system prints a ticket which shows target batch weights and actual batch weights and header data such as a date time code and customer name and address. It then prompts the operator to either recycle the current batch or initiate a new batch.

## COMPUTER DESIGN

The design process for the controller was performed within the following tasks:

1. Processor architecture and external interface design.
2. Software design.
3. System integration and testing.

Our design activities in each of these areas are described in the following paragraphs.

### Processor Architecture and External Interfaces

Early in the system design process we chose to implement the controller with an off-the-shelf M-6800 system. The M-6800 was chosen for its versatile I/O structure. The decision to use this system followed a make versus buy analysis which showed overall engineering costs to be minimized through the use of a suitable off-the-shelf mainframe. Following a detailed investigation of candidate M-6800 systems, we chose the GIMIX mainframe for the following reasons:

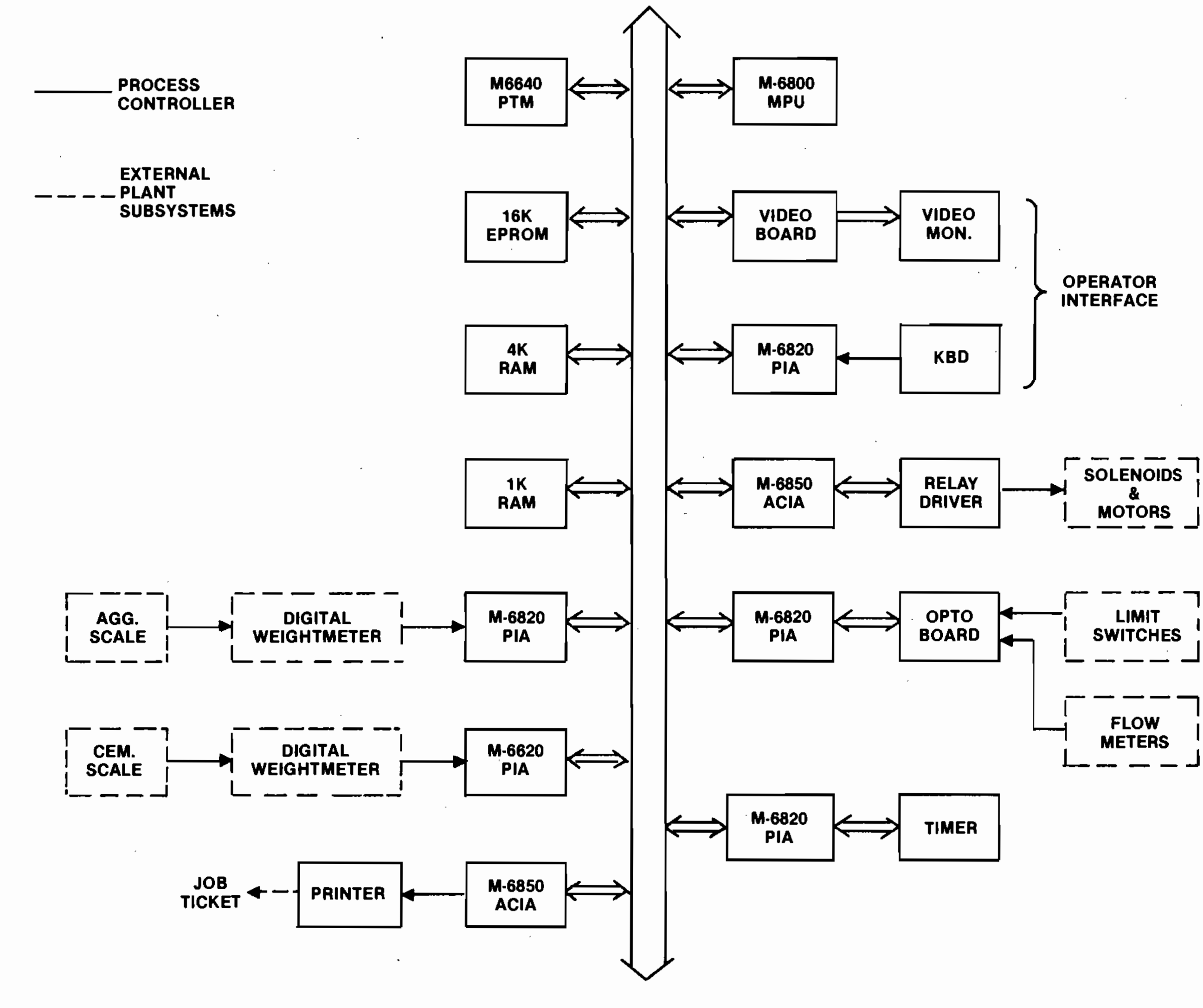


**Figure 2. System Architecture**

1. The GIMIX system is built to the highest quality commercial standards. This minimizes life cycle costs for the controller.
2. The GIMIX power supply offers the best overall AC line regulation for operation within the hostile industrial control environment.
3. The GIMIX relay driver board and opto board are ideally suited as the primary control system interface elements for this application.

The computer hardware configuration for the controller and its external interface is illustrated in Figure 2. The AC power switching to actuate plant functions is performed by the relay driver board. This unit receives computer initiated switching commands via the ACIA port and controls the switching of General Electric RR-8 relays. The various elements of the power relay array interface with solenoids for opening and closing of mechanical material inlet gates and with motors for controlling fluid pumps.

The opto board interfaces with external limit switches associated with mechanical gate openings and closures. This enables the AC switching subsystem to be closed loop in that failure to detect a gate opening or closure following a switching command, constitutes an external plant error condition. Upon detection of the error the system executes an error processing mode.

The scale subsystem components are Electroscale Inc. DR-525 digital weighmeters. Each of these interface with the processor via a PIA digital I/O port. During a weigh-in process, BCD representations of aggregate and cement weights are read into the system from these units.

Flow meters are used to measure the accumulated volume of fluid during admixture and water input process. These units measure volume delivered in terms of successive switch closures. The output of each flow meter is read into the processor through one port on the GIMIX opto board. This enables a special software routine to count the number of successive detected switch closures during an admixture input process. Hence, when the detected fluid volume reaches its target value, a shut off command is sent to the pump motor via the relay driver board.

Certain functions within the process require elapsed time measurement. The GIMIX programmable timer which resides on their eight-port PIA board was selected for this function. This enables software control of the timer and precise measurement of elapsed time.

The system also requires an accurate measurement and recording of the time of day at the completion of each batch cycle. This is performed very nicely by the M-6840 PTM which is on the GIMIX CPU board. The master control program resides on two 8K EPROM boards. This eliminates the necessity for floppy disk drives or tape drives which are susceptible to failure within this dusty industrial environment.

The system also uses 4K of volatile RAM for temporary [s]torage of data values and scratch pad memory. In addition, [w]e included 1K of non-volatile RAM for permanent storage [of] mix design data.

**[S]oftware Development**

All software development was performed on a GIMIX sys[te]m equipped with dual Smoke Signal Broadcasting mini [fl]oppy disk drives. The SSB external storage subsystem was [se]lected for the inherent reliability of the floppy disk con[tr]oller board. System software tools included GMXBUG, [S]SB DOS68.31, the SSB editor/assembler and the Micro[wa]re Systems Inc. ABASIC compiler.

All software components which handle interfaces with ex[te]rnal subsystems were written in M-6800 assembly lan[gu]age. These include the relay driver and opto handlers, the [di]gital weighmeter handler and the elapsed time controller. [T]he main control program which handles operator I/O and [co]ntrols the batching sequence was written in ABASIC. We [se]lected the Microware Systems ABASIC compiler for this [a]pplication since it produces efficient machine code, which is [re]adily adaptable to the EPROM environment. The use of [th]is compiler was a major factor in the success of this project.

The system uses segments of GMXBUG for initialization [an]d operator I/O. We performed minor modifications to [G]MXBUG in order to post special screen formats during sys[te]m initialization.

We paid particular attention to the design of the display [an]d operator interface segment of the software since the [w]eigh batching system must be operated by personnel who [ar]e not familiar with computer operations. To achieve our [de]sign goal of having a simplified operator interface we [ch]ose a tree structured menu system which enables the [op]erator to select a particular processing mode from a dis[pl]ayed menu of processing options. Also during the data in[pu]t mode, the system displays all mix design parameters [fo]llowing operator key in of the data. This presents the [op]erator with a summary of the data which he has entered in[to] the system and enables him to make corrections before [th]e system enters the autobatch mode.

**[S]ystem Integration and Testing**

The weigh batching unit prototype hardware which is illus[tra]ted in Photo 1 has been installed in the Midstate Ready [M]ix Inc. plant in Oneida, New York. It has been successfully [int]egrated with plant electro mechanical controls and has [un]derdone extensive testing for the past several months. The [te]st results show that the system is capable of holding ce[me]nt batching tolerances within one percent of target [we]ights over a target range from 600 to 10,000 pounds and [ag]gregate tolerances within two percent of target weights [ov]er a target range from 1,000 to 30,000 pounds.

The unit also serves as a useful information system for the [ce]ment plant operator in the daily operations.

## [C]ONCLUSIONS

The success of this project demonstrates how a relatively [in]expensive off-the-shelf microcomputer system can be struc[tur]ed to perform a set of real time process control functions [in] a difficult industrial control environment. Development of [thi]s controller was performed within the constraints of a low [en]gineering budget. The key ingredients of this successful [pr]oject were:

1. Application of good systems engineering and computer engineering techniques in the translation of detailed requirements into an efficient structure for the controller.
2. The selection of reliable computer hardware which can operate within this hostile industrial control environment.
3. The selection and judicious use of sound software development tools.☐