SECTION 1

99/8 CHIP SET

1.1 OVERVIEW

The 99/8 requires a considerably larger memory space than the 99/4A has if the /8 was to be the grand and glorious machine it was supposed to be. The 0371 Mapper chip performs this function, and this gives rise to two different memory spaces. The memory space on the uP side is termed the Logical Address Space (LAS), and the space on the other side is called the Physical Address Space (PAS). The size of the LAS is set by the uP, and is a 64K BYTE space (which is the same as that for the 99/4). The PAS is set by the Mapper design, and we have designed it to be 16M BYTEs. You may want to visualize this as two hundred and fifty-six 64K BYTE spaces; so it should be big enough to last a while. We have no idea how one might use this much memory space, but somebody will figure a way to effectively use a bunch of it.

There are some functions that, for 99/4 software compatibility reasons, were based (connected) in the LAS. The block diagram of the /8 shows these, and they are as follows.

* 8K x 8 ROM
* 2K x 8 RAM
* VDP
* Sound Generator
* P-CODE Libraries
* Speech Synthesizer

The functions to the right of the Mapper on the /8 Block Diagram except for the System GROMs (and those in the Command Module) on the /8 Block Diagram are based in the PAS. The Mapper chip was implemented in a "Standard Cell CMOS" process. It as of 7/7/83 the only CMOS device in the /8. The other custom chips to be discussed here are Low Power Schottky Gate Arrays.

1.2 AMIGO-0371

The 0371 Mapper chip is responsible for a number of different things, but the most important is generating a 24-bit address out of the 16-bit uP address presented to it by the uP. It does this by splitting the 16-bit address into two quantities. The Most Significant (MS) 4-bits are used to select a 24-bit address register (more correctly termed a base register) within the Mapper chip. There are 16 of these base registers available in the Mapper chip.

The remaining 12-bits of the uP address are added to the 24 bit contents of the selected base register (the uP portion is added to the Least Significant (LS) end of the Mapper 24-bit quantity) to form the full 24-bit PAS address.

The Mapper chip and I/O Port are both "pin limited", and the 24-bit address is put out in a multiplexed form. See the detailed discussion of the Mapper chip for an in depth explanation.

At one time in our hectic life in the PCC we had to maintain a given amount of Hardware compatibility with the 99/4A. This has since degenerated to cover only the Command Module Port, but logic to make design life easier in the /4A Expansion Box still exists in the Mapper chip. We generate a TMS9900 like set of memory control signals (PMEMEN*, PH3*, PWE*, and PDBIN) as our uP in the /8 is timed differently than the 9900. This turned out to be a real pain because there is no phase relationship between our TMS9537 uP input and output clocks. We will use either a Phase Lock Loop (PLL) or TTL propogation delays to handle this irregularity. Neither is a desirable solution. On top of all of this, the TMS9901 will not reliably function (won't malfunction all of the time) with the 9537 square wave clock. This was another reason for the 25% active low duty cycle PH3* clock.

It seemed like the thing to do at the moment to generate a 320 KHz clock to drive the Speech Synthesizer with. The Mapper provides this constant frequency; thus a "husky masculine voice" should be obtained all of the time.

The Mapper chip is responsible for providing the Chip Select for for the 2K x 8 SRAM in the LAS and the 64K Byte DRAM in the PAS.

As you can see by now, the Mapper chip is kind of a garbage chip. It was an easy thing to add these extras because the Data Bus, the LAS and the PAS all go to this 64-pin package.

1.3 VAQUERRO-CF40063

Vaquerro is responsible for chip selecting all of the devices in the LAS BUT the 2K x 8 SRAM. Our uP is considerably faster than the 9900, and we must provide additional logic to see to it that Software allows a sufficient inactive time for both GROM chips and the VDP chip. Software that observes these rules in the 99/4A clearly violates them when run on our uP. In fact, Munch Man eats the maze sides because we accessed the VDP chip too often (did not allow an 8 us inactive time from the end of one access to the beginning of another).

1.4 POLLO-CF40064

Our naming of the chips took a turn for the worse at this point in time because I thought the need for code names was somewhat unnecessary. The "chicken" chip is a DRAM controller that is used to control the 64K BYTE DRAM in the PAS. I was told that I could use nothing but 16K x 4 DRAM chips that didn't fit too good (instead of the 64K x 1 chips that fit real good); so I got cute and designed POLLO to function in either the /8 or in the /4 32K Memory Expansion for the Expansion Box. I also added the capability of using 64K x 1 chips if management should change their "will never change" mind. I did, and they did, and for once things turned out right. I see this chip being changed in the future to accomodate perhaps 256K x 1 chips or perhaps two banks of 64K x 1 chips (a 128K BYTE on-board DRAM). As of 7/7/83 this effort is neither defined not under way.

1.5 OSO-CF40065

The bear (OSO) is the HEX BUS controller chip, and it interfaces to the /8 just like a memory chip does. The HEX BUS is a 4-bit data bus with two status/control lines. OSO accepts data in 8-bit chunks, and sends it four bits at a time down the bus. It must get two 4-bit nibbles before it will tell the uP it has a BYTE of data for it. This automatic control of the bus is the big difference between OSO and the other HEX BUS controller chips.

1.6 MOFETTA-CF40066

MOFETTA (skunk) is another garbage chip that is mainly responsible for chip selects in the PAS (for OSO, etc). It has a

screw up in the logic going from our schematics to their 4-NAND gates (in the Gate Array), and it does not control OSO as it should; thus, the July 1983 build will not be a fully functional system. Fixed chips will be available for the September 1983 build for sure, and maybe for the August 1983 build.

In addition to the chip selects, Mofetta is responsible for resetting the /8 when a Command Module is just plugged in (but not when one is removed). Contact debounce encountered when a Command Module is either inserted or removed is automatically filtered within the chip, and resistors and diodes external to the chip interface pin #1 on the Command Module to Mofetta.

Since the 9118 VDP does not supply the 447+ KHz GROM clock as does the TMS9918, Mofetta divides the 10.7 MHz clock by 24 to obtain the GROM clock. In addition to this Mofetta also controls the 32K BYTE ROM. There are two address bits that must be manipulated because the ROM is not "linear" in the PAS. The first 16K is linear, but the other two 8K sections function as DSR ROMs, and must be paged in and out.

The /8 differs from the /4 in that a hardware reset may be generated with software. This insures a hard system reset from the inside of the machine on out through the I/O Port. While we are on the subject of resetting, the reset line that goes through the I/O Port is bidirectional. A hardware reset may be affected by other means (not implemented by us at this time), and fed back into the /8 through the I/O Port to obtain the same reset as does either a Command Module insertion or a Software reset.