PicoGAME HDMI

PCB version 1.0
Document version 1.1

Mixtel 90

This is a natural progression from the VGA versions of PicoGAME. It is designed to be largely compatible but there are some necessary changes because there are now fewer available GPIO pins. However, the system is still very flexible.

This is not really a beginners constructional project as the HDMI socket is a little challenging to solder. It's certainly not impossible though if you are careful and take your time. More info in Appendix 1.

The system uses the same Hammond 1593 enclosure as the previous PicoGAME devices, but this time it is possible to build it without having to cut holes in the sides.

There are two PCBs used, both are 100mm square or less. The first is the main PCB, the second contains front and rear panels, a daughter board holding the SD card socket and LED, a second daughter board for the Reset button and two small adapter boards which can be used to connect WII controllers to the DE9 sockets.This PCB has to be cut up by the constructor. It is possible to construct the system without the second PCB. There are positions on the PCB to mount a Reset button and micro SD card module with access from the right hand side.

Specification

- Uses the Raspberry Pi Pico 2 module with PicoMite firmware by Geoff Graham and Peter Mather.
- HDMI video output
- Internal USB hub with four sockets on the rear
- USB-C connector for 5V power and console connection
- Linear regulator for 3V3 power
- DAC audio output (8-bit, 10-bit or 12-bit resolution) via a 3.5mm jack socket on the front
- Real time clock (which doesn't use the analogue pins on Port A this time!)
- Two DE9M controller ports on the front panel
- Full size SD card socket on the front panel
- Front panel LED which can be Power On or replace the Heartbeat LED (or both!)

Several variations of joysticks can be used:

Atari 800/VCS has switches, Pad A and Pad B. There is a single Fire button as above.

Atari ST is switched only. It has an additional button on pin 9

Commodore Amiga, Vic20, C64 & C128 have switches, Pot X (Button 2) & Pot Y (Button 3). Pin 6 is Button 1.

Do not use joysticks intended for the built-in ports on the Sinclair Spectrum +2, +2A or +3. These are wired differently and are incompatible.

Both controller ports can also handle NES/SNES controllers (the type that have DE9F plugs) and I2C controllers.

The two controller ports are connected as follows:
Note: Port A pin 6 used to be GP14 but this is no longer available.

Game port A DE9M socket:

		Controller mode	Paddle / Analogue stick mode	Switched Joystick mode	I2C port mode
1	GP0	n.c.	n.c.	Up	n.c.
2	GP1	Data	n.c.	Down	n.c.
3	GP2	Latch	n.c.	Left	n.c.
4	GP3	Clock	n.c.	Right	n.c.
5	GP26	n.c.	Pad B / Pot Y	(See Commodore)	I2C2 SDA
6	GP23	3V3	Fire / Trigger	Fire / Trigger	n.c.
7	3V3	n.c.	3V3	3V3	3V3
8	GND	GND	GND	GND	GND
9	GP27	n.c.	Pad A / Pot X	See Atari ST & Commodore)	I2C2 SCL

Game port B DE9M socket:

		Controller mode	I2C port mode
1	n.c.	n.c.	n.c.
2	GP28	Data	n.c.
3	GP2	Latch	n.c.
4	GP3	Clock	n.c.
5	GP4	n.c.	I2C SDA - System I2C
6	n.c.	3V3	n.c.
7	3V3	n.c.	3V3
8	GND	GND	GND
9	GP5	n.c.	I2C SCL - System I2C

The 3V3 supplies on Pin 7 are not short-circuit protected (other than any protection given by the regulator). This allows built-in electronics such as rapid fire circuits to have the best chance of operating. The Pin 6 supplies pins are current limited to 22mA each. as these will be shorted to GND by Fire / Trigger buttons. The joystick switch inputs are active-low.

Port B functionality is a little limited, but it is possible to use a second NES or SNES controller in it, using common CLOCK and LATCH signals and GP28 as the DATA signal. Port B also now carries the System I2C connections so it can be used for suitable controllers such as the WII Classic.

If you require a COM port connection then the only option now is to use Port A pin 1 (COM1 TX) and pin 2 (COM1 RX).
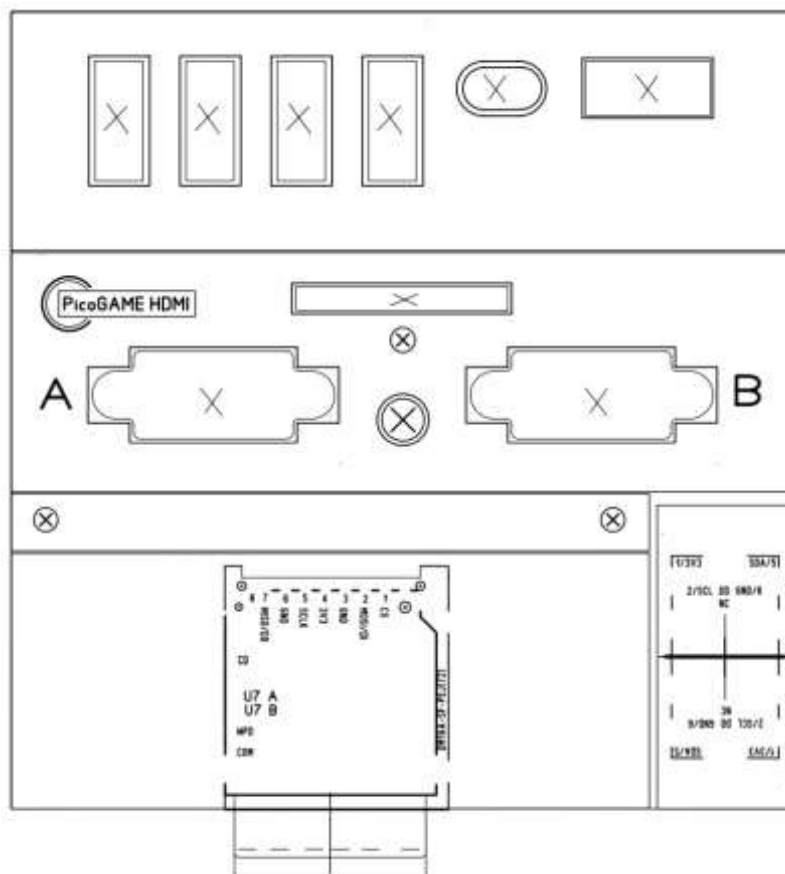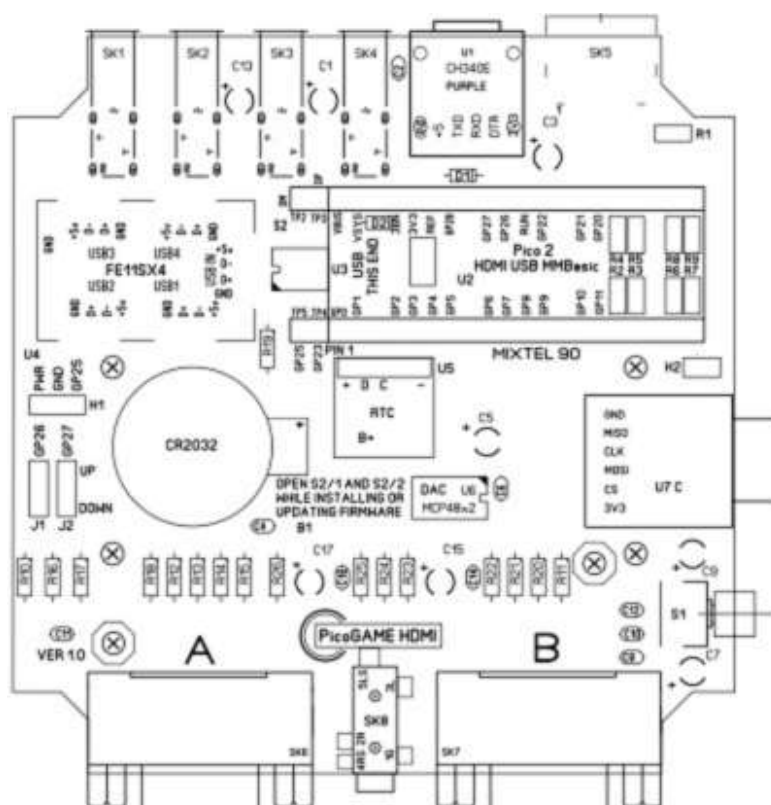
Customising the PicoMite

The Pico has 22 male header pins fitted to each side rather than the usual 20. This allows GP25, GP23, DM and DP to be connected via the usual connectors. The finished module is still breadboard compatible and the modification is easily removed.
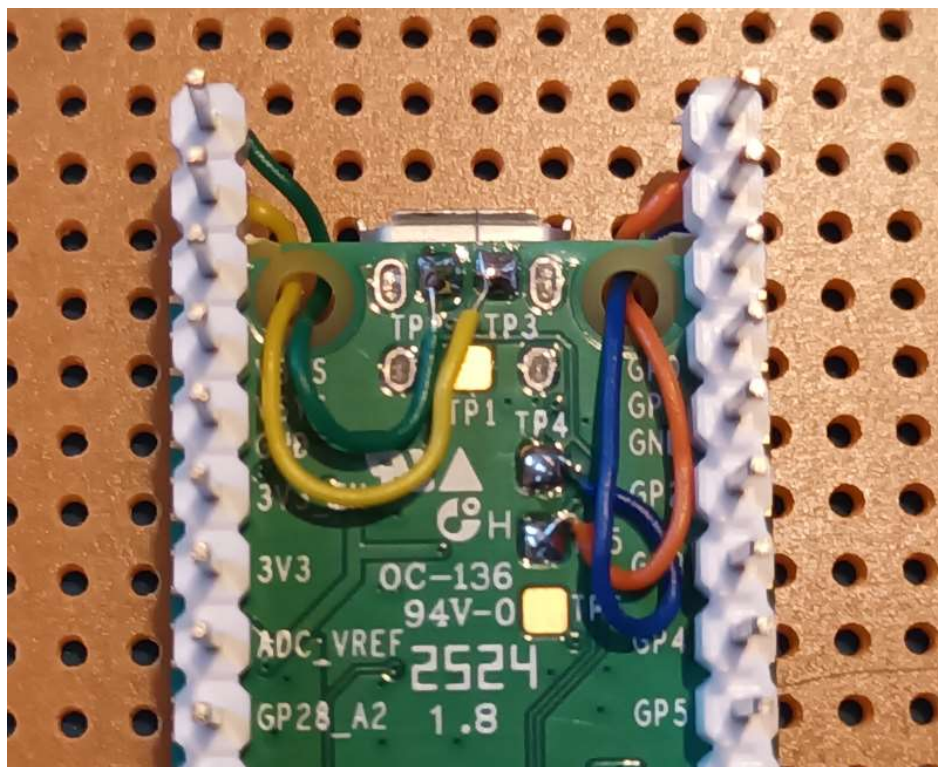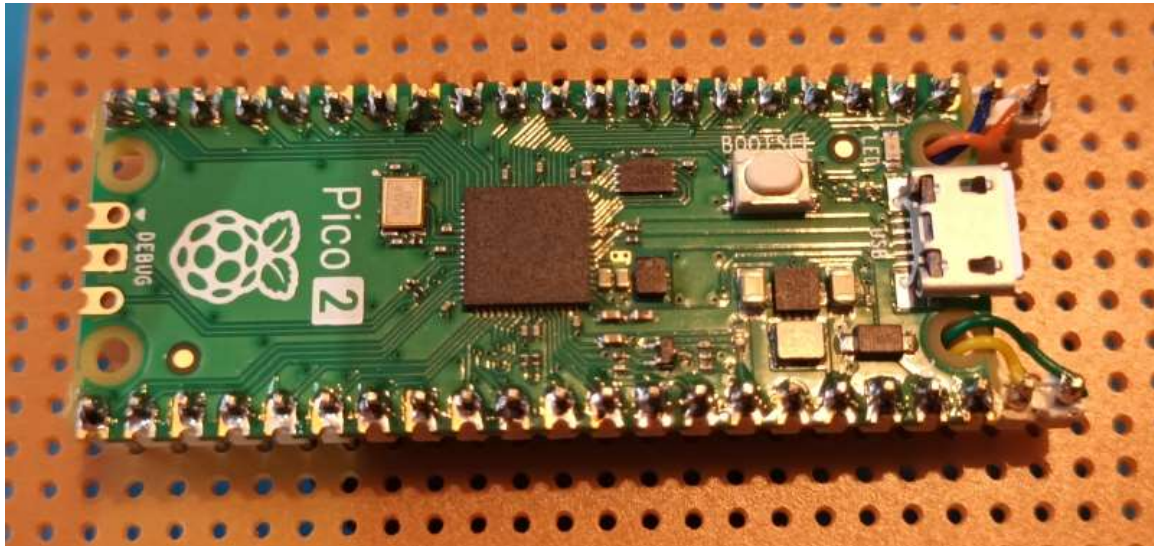
It is possible to build PicoGAME HDMI without this modification. The changes are as follows:
Omit S2. DM and DP have to be on a small micro USB plug wiich is wired to the D+ and D- pads on the USB hub. This is unplugged during MMBasic installation and upgrading. As the micro USB plug obstructs S2 this cannot be used.even if fitted.
There is no access to GP25 so the front panel LED cannot indicate the Heartbeat signal.
There is no access to GP23 so Port A cannot have a Fire button on pin 6.

Bill of Materials

Name	Value	Comment
PCB		custom
Enclosure		Hammond 1593WBK enclosure (optional) - Mouser 546-1593WBK, Digikey 164-1593WBK-ND, RS 228-7399

(White and Translucent blue options also available e.g. RS white 229-1681, blue 228-7400)

Name	Value	Comment
B1	CR2032	cell and holder - optional
C1	10uF	tantalum electrolytic capacitor
C2	100n	Ceramic cap
C3	22uF	tantalum electrolytic capacitor
C4	100n	Ceramic cap
C5	10uF	tantalum electrolytic capacitor
C6	100n	Ceramic cap
C7	10uF	electrolytic capacitor
C8	100n	Ceramic cap
C9	10uF	electrolytic capacitor
C10	100n	Ceramic cap
C11	100n	Ceramic cap
C12	100n	Ceramic cap
C13	22uF	electrolytic capacitor
C14	100n	Ceramic cap
C15	47uF	electrolytic capacitor
C16	100n	Ceramic cap
C17	47uF	electrolytic capacitor
D1	1N5158	or similar 1A Schottky diode
H1	3-way	male header pins
H2	2-way	male header pins
J1	3-way	male header pins with jumper
J2	3-way	male header pins with jumper
R1	10K	1206 resistor
R2	270R	1206 resistor
R3	270R	1206 resistor
R4	270R	1206 resistor
R5	270R	1206 resistor
R6	270R	1206 resistor
R7	270R	1206 resistor
R8	270R	1206 resistor
R9	270R	1206 resistor
R10	1K	resistor
R11	2R2	resistor
R12	10K	resistor
R13	10K	resistor

R14	10K	resistor
R15	10K	resistor
R16	10K	resistor
R17	10K	resistor
R18	150R	resistor
R19	10K	resistor
R20	150R	resistor
R21	120R	resistor
R22	1K	resistor
R23	10K	resistor
R24	120R	resistor
R25	1K	resistor
R26	10K	resistor
R27	1K	resistor

S1	B3F-315n	Horizontal 6x6 tactile switch c/w B32cap
S1B	B3F-315n	Horizontal 6x6 tactile switch
S2	2, 3 or 4 way	DIP switch (only 2 ways are used)

SK1	USB skt	FCI 73725-0110BLF (RS 771-0048), Mouser 649-73725-0110BLF
SK2	USB skt	FCI 73725-0110BLF (RS 771-0048), Mouser 649-73725-0110BLF
SK3	USB skt	FCI 73725-0110BLF (RS 771-0048), Mouser 649-73725-0110BLF
SK4	USB skt	FCI 73725-0110BLF (RS 771-0048), Mouser 649-73725-0110BLF
SK5	HDMI skt	
SK6	DE9M	TE Connectivity Amplimite 5747840-6, RS446-5209, Toby Electronics DMR09P
SK7	DE9M	TE Connectivity Amplimite 5747840-6, RS446-5209, Toby Electronics DMR09P
SK8	audio jack	RS 705-1490

SP1	M3 x 15mm	FF plastic spacer with screws
SP2	M3 x15mm	FF plastic spacer with screws

U1	USB-C skt	USB-C - TTL converter
U2	LM1117	3V3 voltage regulator SOT-223
U3	Pico 2	with pins and MMBasic. See modification.
U4	FE11SX4	USB 2.2 module
U5	RTC	mini RTC - for Raspberry Pi
U6	MCP48x2	dual DAC with 8-pin DIL socket - see notes
U7A	DM1AA-SF-PEJ(72)	Push-push SD card connector
U7B	XF-40T	Push-pull SD card connector
U7C		micro SD card module

Notes:

Most of the components are available from sources such as AliExpress and ebay.

The PCBs are 1.6mm thick. This thickness gives rigidity as some of the fixing points are in awkward positions. It's also ideal for the front and rear slots in the enclosure. The positions of the holes in the front and rear panels has been calculated based on this thickness.

S1 and S1B are the same type of switch. S1 has a cap and is for fitting to the main PCB. S1B has no cap and is for fitting at the rear. Only one is required. If mounting the Reset button in the rear position you will need to use a countersunk screw above the USB sockets as there is very little clearance here.

This project is designed around the official Pico 2 only. It is almost certainly incompatible with anything else.

If used with the Pico 2 W there are problems as GP23, GP25 and the Heartbeat signal are not available. The board can be reconfigured using two solder blob pads thus:
Change LK1 from S to W
Change LK2 from S to W
The red LED now lights if the wireless is powered up (there is no Heartbeat indication other than on the Pico)
A switch-type joystick on Port A cannot now use the Fire button on pin 6. If necessary you can restore this by closing LK3. This will re-enable the Fire button, but on GP27 instead of GP23. You will lose the use of I2C2.
(Note: MMBasic does not support Bluetooth on the Pico 2 W)

U6 can be one of three different, interchangeable versions:
MCP4802 - 8-bit resolution
MCP4812 - 10-bit resolution
MCP4822 - 12-bit resolution
MMBasic will work with any of these. Best audio quality is with the MCP4822.

U7A, U7B and U7C are alternatives. If U7C is not used fit a row of 6 male header pins in its place.

The 2-colour LED can be any second colour but the side connected to GP25 must be red. This has no series resistor of its own (in normal operation) and, because of the lower Vf, will light instead of the green LED on the Pico. Red/green and red/blue are readily available.

The audio output component values will drive reasonably sensitive headphones. If you prefer to use externally amplified speakers then the output might be too high. You can reduce the value of R23 and R26 to about 4K7 in most cases and increase the value of R22 and R25 to a similar value to reduce the level. None of these values are critical.

The real time clock (RTC) uses a very cheap but quite acceptable module intended to plug onto the GPIO pins of a Raspberry Pi. It is self containe, with a soldered on lithium cell. Unfortunately you have no way of knowing how old these modules are when you get them and the cell has been in use since manufacture. For this reason the PCB includes space for a CR2032 cell and holder. Once the built-in cell dies simply unsolder it from the module and connect a length of wire from the positive pad (the one closest to the connector) to the B+ pad on the PCB. Plug the module back in and the bigger, replaceable, cell will keep it running. A link on the pcb allows use of a jumper wire on the RTC Rather than wiring it to the board..
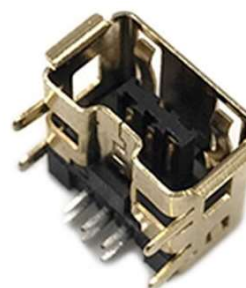
U1

U4

U5

HDMI socket

U7C

WII Controller socket

Assembly

If the second PCB is being used it must be cut into it's individual sections.
Ir S1 (Reset) is to be mounted on the main PCB then a suitable hole will need to be drilled in the side of the case. This is best marked out before the PCB is populated.

Start by mounting th HDMI socket (more info in Appendix 1) , the regulator and the 1206 surface mount resistors. Note that R1 is a different value to the others.

The USB hub is mounted "upside down, with the crystal towards the PCB. Solder short lengths of bare wire into each connection hole so that they project on the components side. Use a simple jig such as the blade of a screwdriver and bend all the wires out, away from the components. The jig must be wider than the height of the crystel so that the board will stand on its "legs". Now trim off excess wire, position the hub over the pads on the PCB and solder them down.
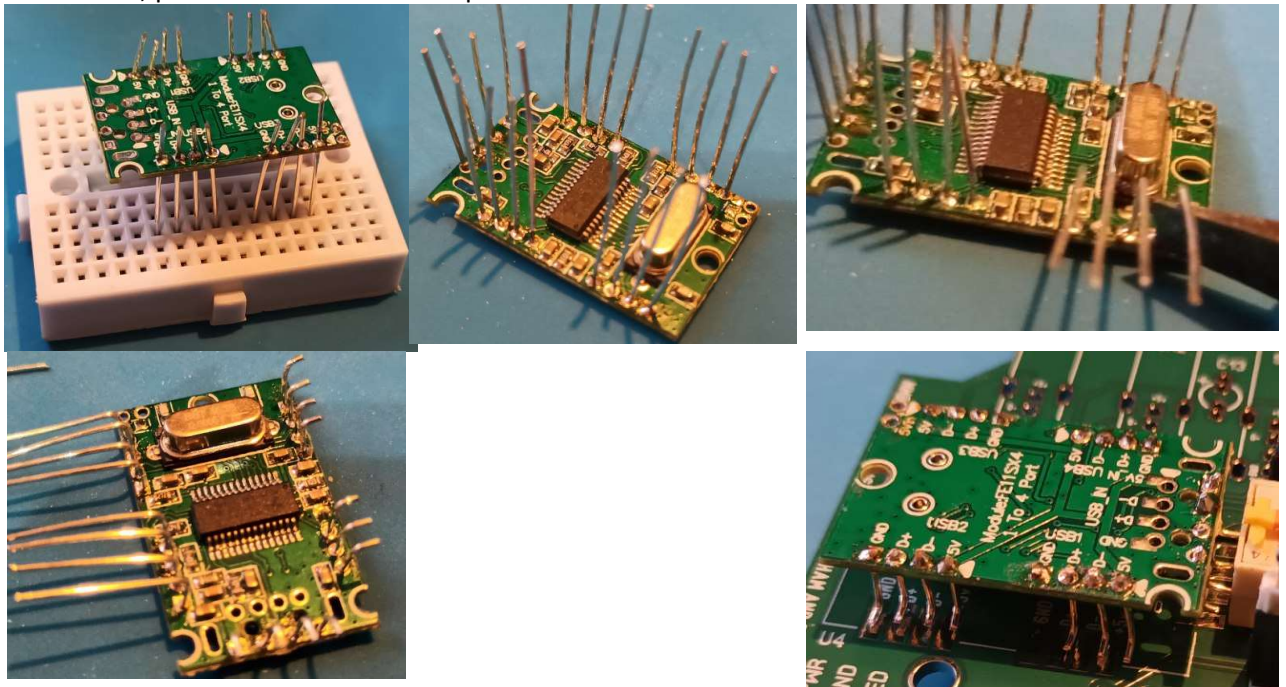
Fit C13, C1, C2, C3, S2, D1 and U1 (the USB-C to TTL converter)

Fit the two 22-way SIP sockets for the PicoMite.

Fit SK1, SK2, SK3 and SK4

You can now fit the remainder of the low level components leaving the DE9 connectors until last.

U8 (the DAC) should be in a socket,

The CR2032 battery and holder are optional. They can be used to extend the life of the RTC module indefinitely.

If you are going to use the daughter PCB with the full size SD card socket and LED on it then fit a strip of 6 male header pins instead of U7C. You will also need a strip of 3 male header pins for H1. The daughter board is fitted on two M3 x 15mm FF spacers with screws from above and below the assembly. It would be a good idea to use shakeproof washers on the bottom screws.

If the Reset button is to be mounted on the rear then there is insufficient depth to use the button cap. The mounting strip is fixed to the lid of the enclosure and two wires with Dupont connectors are used to connect it to H2.

The second PCB includes two small adapter boards for WII controllers. If the correct thickness (1.6mm) of PCB has been used then these will fit between the solder-bucket connections of DE9F plugs. They are then soldered to pads on both sides of the PCB. The WII socket is then soldered onto the board. Note: MMBasic supports WII controllers on System I2C only (Port B). If you write a suitable handler then the second I2C channel (Port A) can also be used.

Configuration

If the Pico 2 W is to be used then there are two "solder blob" links to change and there are changes to the system operation. Please see the Notes section. The Pico 2 and Pico 2 W are not fully pin compatible in this project and are not directly interchangeable.

Open S2/1 and S2/2. These are numbered from the left hand side.
Set JP1 and JP2 to the UP position.
Connect the Pico directly to the PC and install the HDMI USB version of MMBasic.
When MMBasic has been installed remove the programming lead and close S2/1 and S2/2.
Connect the PC to the USB-C socket and use a terminal program to access the console. The baud rate is 115200 by default.
Set the following options:

OPTION PICO OFF
OPTION HDMI 5, 2, 7, 0
OPTION SYSTEM I2C GP4, GP5
OPTION SDCARD GP10, GP11, GP21, GP20
OPTION AUDIO SPI GP22, GP6, GP7
OPTION RTC AUTO ENABLE

The HDMI display and USB keyboard and mouse will now be operational.

APPENDIX 1 - Fitting the HDMI connector

As mentioned previously this can be a little challenging. There are other ways of approaching this but my current favourite technique is as follows:

Materials used:
Solder paste, Gel flux, Isopropyl alcohol, cotton bud, toothpick, magnification, soldering iron with pointed bit, fine desoldering braid.

Get a good magnifier ready. If your eyes are anything like mine you'll need it! I also use a USB microscope for inspection.

Ensure that the PCB pads and the HDMI pins are clean by wiping them with Isopropyl Alcohol on a cotton bud.

On a non-porous surface squeeze out a tiny amount of solder paste.

Using a toothpick apply solder paste to the PCB surface of each pin of the HDMI connector. It can be a little tricky but it's worthwhile taking your time over this. Don't get too fussy, but try not to apply way too much and keep to the pad end.

Apply a small amount of gel or liquid flux to the PCB pads. You don't need a lot, and the gel is slightly easier to control.

Put the HDMI connector on the pcb and solder one of the support lugs down at each end. Some flux might be useful if they are stubborn.

Using a clean soldering iron tip gently wipe each contact pin of the connector towards you. You need a decent thermal contact so you can apply the tiniest bit of solder to the tip if you wish. You aren't trying to solder the pins down though, just activate the flux and solder that's in the paste. The iron needs to be fairly hot - I tend to use about 300C and only solder for a short time - you can always come back to a pin..

If you get solder blobs between pins they can be removed using the toothpick and/or the solder braid. If you get too much heat too close to the plastic moulding it will melt a bit. Don't get too upset about this but try not to do it. :)

If you get that stage right you'll be rewarded by a very neat set of connections (although you may need to remove flux residue to see them). Use a multimeter to test between every connection, it's adjacent connections and GND. It's a bit fiddly to test to the contacts inside the HDMI connector but it can be done.

Complete by soldering down the remaining support lugs.

After this the SMD resistors are oh so easy! :)