ARM Virtual Generic Interrupt Controller v2 (VGIC)

Device types supported:

- KVM_DEV_TYPE_ARM_VGIC_V2 ARM Generic Interrupt Controller v2.0

Only one VGIC instance may be instantiated through either this API or the legacy KVM_CREATE_IRQCHIP API. The created VGIC will act as the VM interrupt controller, requiring emulated user-space devices to inject interrupts to the VGIC instead of directly to CPUs.

GICv3 implementations with hardware compatibility support allow creating a guest GICv2 through this interface. For information on creating a guest GICv3 device and guest ITS devices, see arm-vgic-v3.txt. It is not possible to create both a GICv3 and GICv2 device on the same VM.

Groups:

KVM_DEV_ARM_VGIC_GRP_ADDR

Attributes:

KVM_VGIC_V2_ADDR_TYPE_DIST (rw, 64-bit)
Base address in the guest physical address space of the GIC distributor register mappings. Only valid for KVM_DEV_TYPE_ARM_VGIC_V2. This address needs to be 4K aligned and the region covers 4 KByte.

KVM_VGIC_V2_ADDR_TYPE_CPU (rw, 64-bit)
Base address in the guest physical address space of the GIC virtual cpu interface register mappings. Only valid for KVM_DEV_TYPE_ARM_VGIC_V2. This address needs to be 4K aligned and the region covers 4 KByte.

Errors:

-E2BIG	Address outside of addressable IPA range
-EINVAL	Incorrectly aligned address
-EEXIST	Address already configured
-ENXIO	The group or attribute is unknown/unsupported for this device or hardware support is missing.
-EFAULT	Invalid user pointer for attr->addr.

KVM_DEV_ARM_VGIC_GRP_DIST_REGS

Attributes:

The attr field of kvm_device_attr encodes two values:

```
bits:    | 63  .... 40 | 39 .. 32 | 31  ....  0 |
values:  |   reserved  | vcpu_index |     offset    |
```

All distributor regs are (rw, 32-bit)

The offset is relative to the "Distributor base address" as defined in the GICv2 specs. Getting or setting such a register has the same effect as reading or writing the register on the actual hardware from the cpu whose index is specified with the vcpu_index field. Note that most distributor fields are not banked, but return the same value regardless of the vcpu_index used to access the register.

GICD_IIDR.Revision is updated when the KVM implementation of an emulated GICv2 is changed in a way directly observable by the guest or userspace. Userspace should read GICD_IIDR from KVM and write back the read value to confirm its expected behavior is aligned with the KVM implementation. Userspace should set GICD_IIDR before setting any other registers (both KVM_DEV_ARM_VGIC_GRP_DIST_REGS and KVM_DEV_ARM_VGIC_GRP_CPU_REGS) to ensure the expected behavior. Unless GICD_IIDR has been set from userspace, writes to the interrupt group registers (GICD_IGROUPR) are ignored.

Errors:

-ENXIO	Getting or setting this register is not yet supported
-EBUSY	One or more VCPUs are running
-EINVAL	Invalid vcpu_index supplied

KVM_DEV_ARM_VGIC_GRP_CPU_REGS

Attributes:

The attr field of kvm_device_attr encodes two values:

```
bits:     | 63   .... 40 | 39 .. 32 | 31   ....   0 |
values:   |   reserved   | vcpu_index |     offset    |
```

All CPU interface regs are (rw, 32-bit)

The offset specifies the offset from the "CPU interface base address" as defined in the GICv2 specs. Getting or setting such a register has the same effect as reading or writing the register on the actual hardware.

The Active Priorities Registers APRn are implementation defined, so we set a fixed format for our implementation that fits with the model of a "GICv2 implementation without the security extensions" which we present to the guest. This interface always exposes four register APR[0-3] describing the maximum possible 128 preemption levels. The semantics of the register indicate if any interrupts in a given preemption level are in the active state by setting the corresponding bit.

Thus, preemption level X has one or more active interrupts if and only if:

$$APRn[X \bmod 32] == 0b1, \text{ where } n = X / 32$$

Bits for undefined preemption levels are RAZ/WI.

Note that this differs from a CPU's view of the APRs on hardware in which a GIC without the security extensions expose group 0 and group 1 active priorities in separate register groups, whereas we show a combined view similar to GICv2's GICH_APR.

For historical reasons and to provide ABI compatibility with userspace we export the GICC_PMR register in the format of the GICH_VMCR.VMPriMask field in the lower 5 bits of a word, meaning that userspace must always use the lower 5 bits to communicate with the KVM device and must shift the value left by 3 places to obtain the actual priority mask level.

Errors:

-ENXIO	Getting or setting this register is not yet supported
-EBUSY	One or more VCPUs are running
-EINVAL	Invalid vcpu_index supplied

KVM_DEV_ARM_VGIC_GRP_NR_IRQS

Attributes:

A value describing the number of interrupts (SGI, PPI and SPI) for this GIC instance, ranging from 64 to 1024, in increments of 32.

Errors:

-EINVAL	Value set is out of the expected range
-EBUSY	Value has already be set, or GIC has already been initialized with default values.

KVM_DEV_ARM_VGIC_GRP_CTRL

Attributes:

KVM_DEV_ARM_VGIC_CTRL_INIT
request the initialization of the VGIC or ITS, no additional parameter in kvm_device_attr.addr.

Errors:

-ENXIO	VGIC not properly configured as required prior to calling this attribute
-ENODEV	no online VCPU
-ENOMEM	memory shortage when allocating vgic internal data